SONDERHEFT 7/1986 öS 100,-/sfr. 14,- Lit. 12000/hfl. 18,-/dkr. 68,- DM 14,-

★HAPPY-COMPUTER★ 4. SCHNEIDER SONDERHEFT

★HAPPY-COMPUTER★

Markt&Technik

DAS GROSSE HEIMCOMPUTER-MAGAZIN

CP/M Plus
- ★ Optimal eingesetzt
- ★ Grundlagen
- ★ Die besten Tips

Disketten-Laufwerke
- ★ Kaufberatung
- ★ Grundlagen

Die besten Listings
- ★ Super-Spiele
- ★ Nützliche Tips & Tricks
- ★ Interessante Anwendungen
- ★ Faszinierende Grafikprogramme

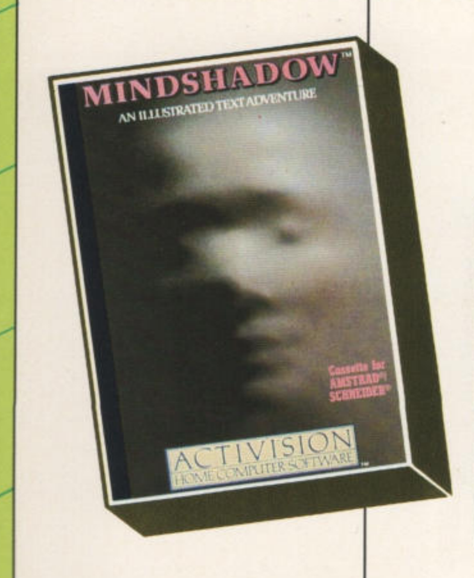
Tips zur Vortex-Karte

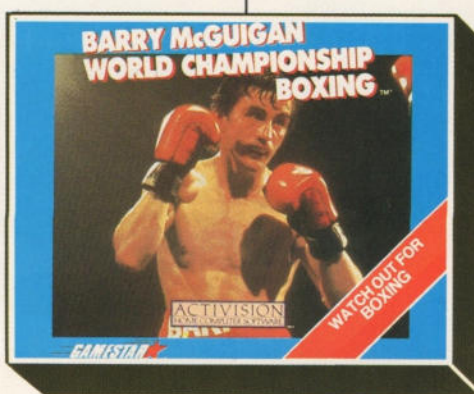
Joyce
- ★ Große Software-Übersicht
- ★ Alle Vorteile des Joyce

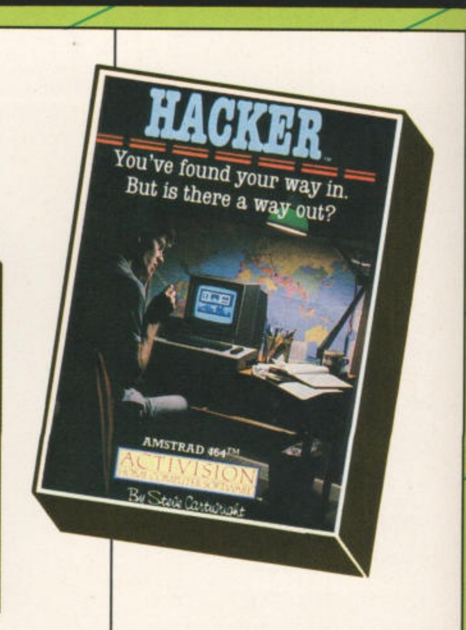


Alle Programme auf Diskette und Kassette erhältlich

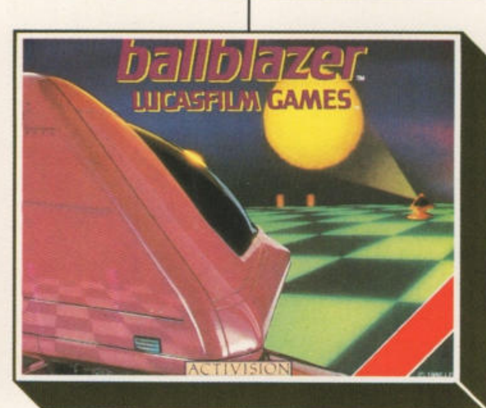
UNSERE BESTEN FÜR DEN SCHNEIDER CPC

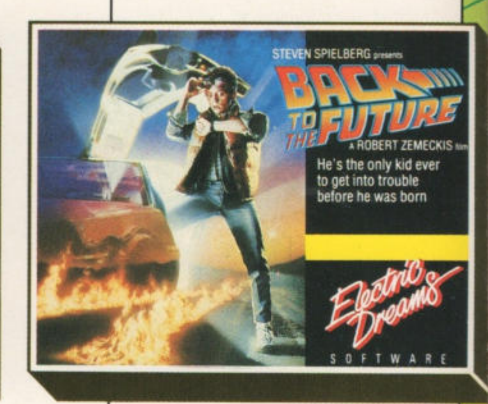


















NEU!

Erhältlich als Kassette für CPC 464, als Diskette für CPC 664/6128





Schneider – im Wandel der Zeit

Joyce heißt der neue Star bei Schneider. Ein halbes Jahr hat es gedauert, bis der für den Geschäftsbereich gedachte Computer auf Platz eins der Schneider-Verkaufshitparade vorrückte. Nicht zuletzt der drastische Preissturz auf 1799 Mark hat diese Entwicklung hervorgerufen.

Der Name Joyce steht für einen Computer, der bewährte Technologie zu einem günstigen Preis bietet. Wie von Schneider gewohnt, ist das Angebot wieder rundherum komplett. Monitor, Drucker, Diskettenlaufwerk, Computer und Software, nichts fehlt zum Einstieg in die elektronische Datenverarbeitung. Da man wieder auf bewährte »Bausteine« – wie den Z80 als CPU und CP/M Plus als Betriebssystem, vereint mit modernen Komponenten wie Mallard-Basic und hochintegrierte Schaltungen – gesetzt hat, ist auch der Joyce ein interessantes Gerät.

Aber natürlich bleiben bei diesem Computer wieder eine Menge Fragen offen, sowohl beim Einsteiger als auch beim Profi. Deshalb haben wir beschlossen, dieses Schneider-Sonderheft (es ist übrigens schon das vierte) um einen Joyce-Teil zu bereichern. Dort finden Sie unter anderem Programme zum Abtippen und eine Einführung in das sehr gute Mallard-Basic. Freunde von CP/M Plus kommen im CP/M-Teil auf ihre Kosten.

Dieser Teil ist natürlich auch für die Besitzer des CPC 6128 gedacht. Denn hier werden alle Hilfsprogramme auf den Systemdisketten zusammenhängend und ausführlich besprochen.

Aber auch die Besitzer der ersten Schneider-Computer haben wir nicht vergessen. Die neuesten Spieletrends hat unser Spielespezialist wieder für Sie zusammengetragen. Spiele-Listings zum Abtippen für die Fleißigen, dazu interessante Anwendungsprogramme und wieder eine ganze »Latte« mit Tips & Tricks.

Trotzdem glauben wir, noch mehr auf Ihre Wünsche eingehen zu können. Aber dazu brauchen wir Ihre Hilfe. Eine große Umfrage mit tollen Preisen soll dabei helfen. Und deshalb eine Bitte an Sie: Machen Sie mit. Denn es lohnt sich doppelt. Zum einen können Sie einen der Preise – beispielsweise ein Diskettenlaufwerk oder eine Speichererweiterung – gewinnen, zum anderen bestimmen Sie, was in zukünftigen Sonderheften und Ausgaben von Happy-Computer für Schneider-Freaks stehen wird.

Dem Heimcomputer-Markt steht ein Wandel bevor wie noch nie zuvor – trotz seiner Geschichte. Die 8-Bit-Computer werden angesichts der neuen 68000er von vielen schon totgesagt. Das mag für alte Geräte dieser Klasse vielleicht zutreffen, aber moderne ausgereifte Konzepte machen auch den Z80 weiterhin aktuell. Und neben den 68000ern dringen immer mehr die IBM-kompatiblen 16-Bit-Computer in die untere Preisklasse ein. Die Vielfalt der Prozessoren und damit der Geräte ist noch lange gesichert. Und in dieser Klasse wird Schneider mit seinen Computern auch in Zukunft ein Wörtchen mitreden.

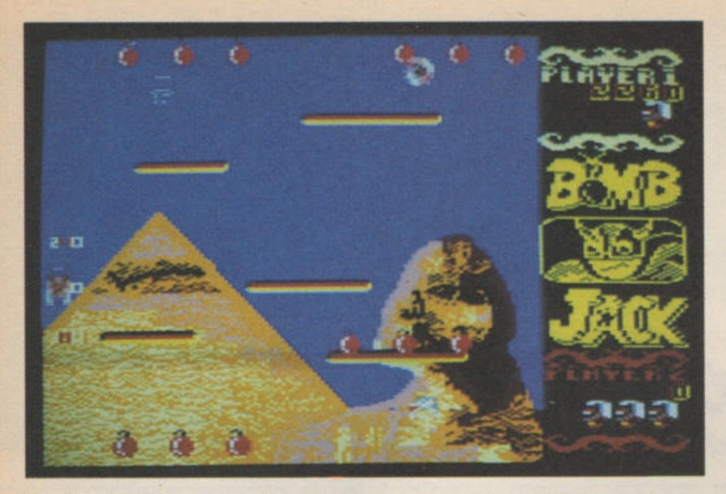
Nicht zuletzt der Kauf von Sinclair durch Amstrad, dem englischen Entwickler der Schneider-Computer, zeigt, daß viele diesem Markt noch eine große Zukunft geben. Auch Vortex, der bekannteste Peripheriehersteller für Schneider-Geräte, setzt auf diese Welle. Denn die Versorgung mit Neuentwicklungen aus Neuenstadt ist gewährleistet.

Mit diesem Heft liefern wir Ihnen wieder mehr als 120 Seiten Informationen an die Hand. Ich möchte an dieser Stelle allen danken, die uns ihr Wissen zur Verfügung gestellt haben. Denn nur mit Ihrer Mitarbeit können wir auf »alle« Ihre Fragen eine Antwort finden.

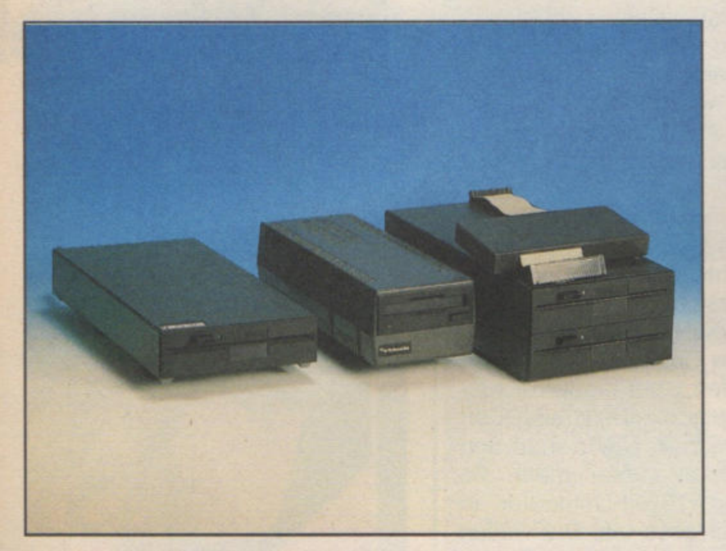
Also machen Sie mit und schicken Sie uns Artikel, Informationen, Anregungen, Listings oder was Ihnen sonst noch einfällt.

Ihr
Andreas Hagedorn

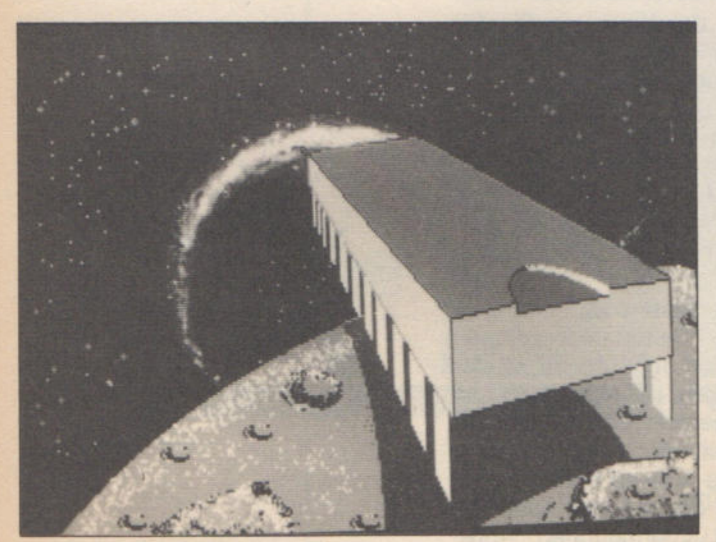
Inhalt



Zwei heiße Renner stellen wir in unserem Spieletest vor: Bomb-Jack, der quer durch die Welt Bomben aufsammeln und entschärfen muß. Es entwickelte sich zum Favoriten bei uns in der Redaktion und begeistert sicherlich alle, die Aktion lieben. **10**



Diskettenlaufwerke war beim Schneider schon immer ein wichtiges Thema. Ob Schneider- oder Vortex-Station, 3-Zoll- oder 5¹/₄-Zoll-Doppel- oder Einfachlaufwerk, lautet die Frage. Bei dieser Entscheidung wollen wir Ihnen behilflich sein. **6**



Auf Ihrer CP/M-System-Diskette finden Sie Dienstprogramme en masse. Das Handbuch geht aber nur wenig darauf ein und läßt viele Fragen offen. Unser CP/M-Teil hilft Ihnen dabei, mit Ihrer CP/M-Version optimal zu arbeiten. **120**

Tips zur Vortex-Karte

CP/M Plus optimal eingesetzt

Grundlagen

Die besten Tips

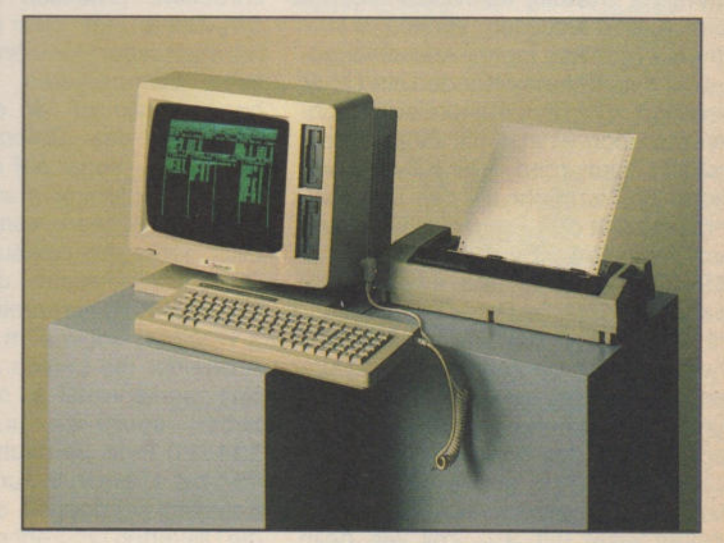
Vermischtes



Es ist noch kein Beethoven vom Himmel gefallen, aber unser Grundlagen-Artikel über Soundprogrammierung kann Ihnen einen Schritt weiterhelfen. Daneben helfen Ihnen weitere Beiträge beim Einstieg in die Grafikwelt und in die Basic-Programmierung. **33**



Die Welt liegt Ihnen zu Füßen, wenn Sie es in unserem Listing »Feldherr« bis zum Diktator gebracht haben. Es ist dem Brettspiel »Risiko« nachempfunden und bedarf einer ordentlichen Portion an strategischem Talent, um seinen Gegner zu bezwingen. **58**



Der Joyce fand im Büroalltag schon weite Verbreitung. Trotzdem sind sich viele Anwender über seine Fähigkeiten noch nicht im klaren. Vom Umgang mit Mallard-Basic und CP/M Plus bis hin zur Softwareübersicht stellen wir den Joyce im Detail vor. **22**

Freie Auswahl



Das Diskettenlaufwerk von Schneider



5¼ Zoll heißt das Format von Vortex

Der Schneider CPC kann sich inzwischen an einem reichhaltigen Angebot an Diskettenstationen erfreuen. Wir haben die beiden gebräuchlichsten herausgesucht und unter die Lupe genommen.

Es ist nichts Ungewöhnliches im Bereich der Heim- und Personal-Computer, daß für ein Gerät Laufwerke verschiedener Hersteller angeboten werden. Daß aber die Fremdfabrikate der Firma Vortex ein solch riesiger Verkaufserfolg sind, verwundert nun doch. Woher kommt das? Eigentlich gibt es zwei Gründe. Einmal gehört Amsdos, das Diskettenbetriebssystem von Schneider, nicht gerade zu den komfortabelsten und leistungsfähigsten DOS-Versionen auf dem Computermarkt. VDOS, das Vortex Disk Operating System, kann da – besonders in der aktuellen Version 2.0 – schon erheblich mehr. Ausschlaggebend für viele Vortex-Kunden sind aber garantiert die Diskettenpreise. Rechnen Sie doch einmal mit: Eine Vortex-Diskette kann rund 700 KByte speichern, das ist mehr als das Doppelte der Kapazität der Schneider-Disketten (pro Seite 154 KByte im IBM-Format, 169 KByte im CP/M- und Vendorformat sowie 178 KByte im Data-Only-Format). Für eine 3-Zoll-Diskette müssen Sie etwa zwölf bis vierzehn Mark aufwenden. Die ungleich leistungsfähigeren Vortex-Scheiben kosten Sie selbst bei Verwendung der empfohlenen Qualitätsdisketten nur sechs bis sieben Mark. Das Laufwerk von Schneider arbeitet nicht mit $3^1/_2$ Zoll wie beim Atari-ST, Amiga von Commodore oder einige IBM-Kompatiblen, sondern mit dem etwas ungewöhnlichen 3-Zoll-Format. Außer Amstrad hat nur die britische Firma Tatung die Hitachi-Laufwerke dieses Formats in ihren – in Deutschland kaum bekannten – Computer »Einstein« eingebaut.Es gibt aber auch Gründe, die für dieses Format sprechen. Erheblich kleiner als der $5^1/_4$-Zoll-Bruder, kann man die Disketten auch in der Hemden- oder Hosentasche transportieren. Die Disketten besitzen nämlich ein stabiles Plastik-Gehäuse, das Deformationen beim Transport – sogar auf dem Postweg – verhindert. Nimmt man die Disketten aus dem Laufwerk, verschließt ein Metallschieber das Gehäuse, so daß kein Staub oder ähnliches die empfindliche Beschichtung verunreinigt.

Weniger angenehm bekommt der Anwender die bereits angesprochene Speicherkapazität zu spüren. Die Disketten speichern auf jeder Seite 184320 Byte. Je nach Format stehen 154 bis 178 KByte zur Verfügung. Die Kapazität verdoppelt sich, wenn man die Disketten umdreht und den Computer die Rückseite beschreiben läßt. Das ist kein Geheimtip aus der Trickkiste, sondern wird von Schneider und den Diskettenherstellern offiziell empfohlen. Dennoch wird die Umdreherei mit der Zeit lästig, und man sehnt sich nach einem Doppelkopf-Laufwerk.

Auch die vierzig Tracks à neun Sektoren, mit denen die Disketten formatiert werden, sind nicht mehr der letzte Stand der Technik. Die Vortex-Laufwerke formatieren 80 Spuren und benötigen dann die altbewährten $5^1/_4$-Zoll-Disketten in der Ausfertigung DSDD (Double Sided Double Density) mit 96 tpi (Tracks per inch). Diese Disketten sind zwar bekanntlich sehr empfindlich gegen Fingergrabbeleien, dafür aber auch erheblich billiger.

Amsdos contra VDOS

Amsdos liegt in einem 16-KByte-ROM. Da aber der Logo-Interpreter im selben ROM »beheimatet« ist, bleiben in Wirklichkeit leider nur noch acht davon übrig. In diesem Bereich ein komplettes DOS unterzubringen, ist schon eine große Leistung. Zwangsläufig wurden aber Kompromisse eingegangen. So ist das Kopieren und Formatieren von Disketten unter Basic nicht möglich. Der Anwender muß dazu auf die mitgelieferte CP/M-Software zurückgreifen. In Basic vorhanden sind nur die wichtigsten Befehle wie DIR (Inhaltsverzeichnis der Diskette), ERA (Löschen von Dateien), REN (Umbenennen von Dateien), USER (Auswahl des Benutzerbereichs), DRIVE, A und B (Selektion eines Laufwerks). DISK, DISK.IN und DISK.OUT schalten vom

Kassettenrecorder auf die Diskettenstation um, TAPE, TAPE.IN und TAPE. OUT bewirken das Gegenteil. VDOS 2.0 ist da entschieden leistungsstärker. Vortex entfernte den Logo-Interpreter aus dem ROM und implementierte statt dessen andere Routinen. Über RSX-Befehle (»Resident System Extensions«) kann der Benutzer Disketten formatieren (FORMAT), einen Dateischutz vereinbaren (CODE) und Dateiattribute setzen (ATTRIBUT). Die relative Dateiverwaltung, die beim Schneider-Laufwerk völlig fehlt, wird bei VDOS über FILES, OPEN, FIELD, GET, PUT und CLOSE realisiert. Über DERROR kann man auch beim CPC 464 Fehlermeldungen von der Diskettenstation abfangen. Vortex ließ den verbleibenden Platz im ROM nicht ungenutzt und nahm neben diesen diskettenorientierten Befehlen noch eine Reihe weiterer nützlicher Utilities auf. FAST etwa beschleunigt die Bildschirmausgabe im Modus 2 um etwa das Doppelte, wenn der Anwender auf Windows verzichtet. SLOW hebt dies wieder auf. Auch die vom 664- und 6128-Basic bekannten Befehle FRAME, GCHAR (Getchar), GPAPER (Graphics Paper), GPEN (Graphics Pen), MASK und UNMASK findet man als RSX-Erweiterungen wieder.

Alle Maschinensprache-Freunde werden von dem eingebauten Maschinensprache-Monitor sehr angetan sein. In etwa so leistungsfähig wie das CP/M-Programm DDT.COM, verarbeitet er aber Z80-Mnemonics. Er kann Breakpoints in Maschinencode-Programmen verwalten, Speicherbereiche mit Bytemustern füllen, Binärdateien lesen und auf Diskette schreiben, disassemblieren, Speicherbereiche kopieren und sogar Maschinenprogramme Schritt für Schritt (Single Step) ausführen. Der eingebaute Zeilenassembler enttäuscht dagegen etwas. Er versteht zwar den kompletten Befehlssatz des Z80-Prozessors, viel mehr kann er aber nicht. Labels, symbolische Sprünge oder sogar Makros sind Fremdwörter für ihn.

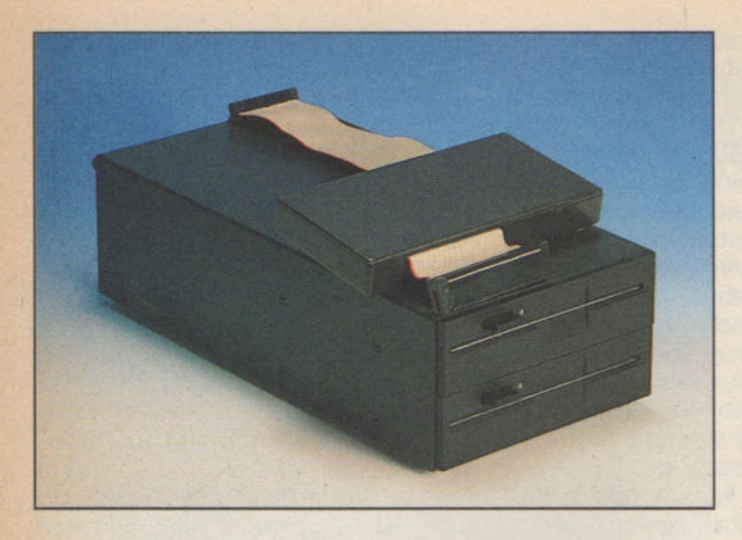
Immer wieder Kompatibilität

Leider ist es aber mit der versprochenen Amsdos-Kompatibilität bei VDOS nicht allzuweit her. Sogar reine Basic-Programme sind nicht frei zwischen beiden DOS-Betriebssystemen auszutauschen. Warum muß denn die Amsdos-RSX »TAPE« bei VDOS gerade »CAS« heißen? Die Tabelle listet die RSX-Erweiterungen unter Amsdos und VDOS auf. Es soll Ihnen helfen, bestehende Programme von einem DOS ins andere umzuschreiben.

Bis vor nicht allzulanger Zeit besaßen Heimcomputer kein Betriebssystem. CP/M oder gar MS-DOS haftete stets der Geruch des Unnahbaren an. Seitdem aber Amstrad und Schneider CP/M auf ihren Maschinen implementiert haben und kostenlos bei der Diskettenstation mitliefern, erlebt es einen ungeahnten neuen Aufschwung. Das unerwartete Comeback von CP/M in einer Zeit der 16-Bit-Computer unterstützte auch der Commodore 128. Bei allen Schneider CPCs gehört das CP/M-Betriebssystem in der weitverbreiteten Version 2.2 zum Lieferumfang der Diskettenstation. Der CPC 6128 verarbeitet zusätzlich noch das weit anwenderfreundlichere CP/M Plus (Version 3.0). Auch Vortex liefert CP/M 2.2 zu seinen Laufwerken aus. Ein großer Teil der CP/M-Standardsoftware ist somit auf dem Schneider-CPC zu verwenden. Wenn auch der Speicher für die CP/M-Programme (TPA = Transient Program Area) mit nicht einmal 40 KByte doch sehr knapp geraten ist. Zusammen mit







Auch ein Doppellaufwerk ist erhältlich

der Speichererweiterung – ebenfalls von Vortex – gewinnt dieser Bereich jedoch noch an Umfang.

CP/M-Programme befinden sich meist auf 5¹/₄-Zoll-Disketten. Hier herrscht allerdings ein wahres Chaos. Jeder Hersteller verwendet sein eigenes Format. Als Besitzer einer 3-Zoll-Diskettenstation bleibt einem nichts anderes übrig, als Firmen zu suchen, die CP/M-Programme in diesem Format anbieten oder in dieses Format konvertieren. Das Angebot nimmt zwar mit der Zeit immer mehr zu, aber den großen Rest der Programme kann man nur auf 5¹/₄-Zoll-Disketten kaufen. Und das zählt zu den eindeutigen Pluspunkten der Vortex-Laufwerke. Vortex liefert nämlich das Programm »PARA«, mit dem der Benutzer eine Vielzahl von CP/M-Formaten lesen und schreiben kann.

Zumindest werden für die 3-Zoll-Formate aber Programme wie Turbo Pascal und Wordstar angeboten. Sie sind inzwischen auch preislich attraktiv geworden, während die Mehrheit der CP/M-Programme sich preislich noch in höheren Sphären befindet. Bei CP/M selbst gibt es keine Unterschiede zwischen den Konkurrenten aus dem Schneider- und dem Vortex-Lager. Denn CP/M 2.2 unterliegt strengen Normen, und Digital Research wacht eisern darüber, daß sich niemand an dem Betriebssystem zu schaffen macht. Den üblichen Satz von Standard-Hilfsprogrammen bieten beide: ASM, DDT, ED, PIP, STAT und wie sie alle heißen. Doch bei den systemspezifischen Utilities finden sich qualitative Unterschiede. Die Schneider-Programme unter CP/M 2.2 erfüllen ihre Aufgabe einwandfrei, glänzen aber nicht gerade durch Benutzerfreundlichkeit. Dagegen strengte sich Vortex bei den neuesten Versionen der Dienstprogramme wirklich an und erstellte ansprechende Software. Die beiden wichtigsten Vortex-Programme heißen Disktool 1.0 und Filecopy 3.0. DISKTOOL.COM erlaubt das Formatieren und Kopieren ganzer Disketten sowie die Übertragung der CP/M-Systemspuren und Parametersektoren zwischen den Disketten. Die Bedienung von Disktool ähnelt der von DISCKIT3.COM, dem Disketten-Dienstprogramm von CP/M Plus auf dem Schneider CPC 6128. FILECOPY.COM bietet dem Benutzer die Möglichkeit, einzelne Dateien oder Dateigruppen zu kopieren. Es ist bereits für jetzige und zukünftige Hardware-Entwicklungen gerüstet und arbeitet auch mit RAM-Disks und Festplatten zusammen.

Preise sind immer ein leidiges Thema. Besonders da der Geldbeutel der Computerfreaks von Natur aus ständig leer zu sein scheint. Deshalb ein kurzer Preisvergleich. Als Einzel-Laufwerk kommt die 3-Zoll-Station von Schneider erheblich billiger als das Vortex-Laufwerk. Wählen Sie allerdings gleich eine Doppelstation, schmilzt der Preisvorteil recht schnell zusammen. Die Konfiguration DDI-1 mit FD-1 kostet nur unwesentlich weniger als das Doppellaufwerk von Vortex. Beachten Sie aber auch, daß der größte Teil der Schneider-spezifischen Diskettensoftware, die aus dem Ausland zu uns kommt, nur im 3-Zoll-Format lieferbar ist. Dagegen setzt Vortex den Vorteil einer um ein Vielfaches höheren Speicherkapazität.

Fazit: Wieder einmal kann Ihnen niemand die Entscheidung abnehmen. Es bieten beide Laufwerke Vorteile, doch muß sich wohl jeder an seinen Bedürfnissen orientieren, welchen Kompromiß er eingehen will.

(Martin Kotulla/Udo Reetz)

Amsdos	VDOS	Erklärung
A	A	Laufwerk A wird Standard-Laufwerk
B	B	Laufwerk B wird Standard-Laufwerk
CPM	CPM	Starten des CP/M-Betriebssystems
DIR	DIR	Ausgabe des Disketteninhalts
DISC	DISC	Umschaltung der Ein- und Ausgabe auf Diskette
DISC.IN	DISC.IN	Diskettenlaufwerk als Eingabemedium
DISC.OUT	DISC.OUT	Diskettenlaufwerk als Ausgabemedium
DRIVE	A oder B	Auswahl eines Laufwerks
ERA	ERA	Löschen von Dateien
REN	REN	Umbenennen von Dateien
TAPE	CAS	Ein- und Ausgabe von/auf Kassettenrecorder
TAPE.IN	CAS.IN	Kassettenrecorder als Eingabemedium
TAPE.OUT	CAS.OUT	Kassettenrecorder als Ausgabemedium
USER	SELECT	Auswahl des Benutzerbereichs
	RESET	Mitteilung des Diskettenwechsels
	FORMAT	Diskettenformatierung
	CODE	Programm- und Dateischutz
	ATTRIBUT	Festlegen von Dateiattributen
	FILES	Relative Dateien: Festlegen der RAM-Puffer
	OPEN	Relative Dateien: Öffnen einer Datei
	CLOSE	Relative Dateien: Schließen einer Datei
	FIELD	Relative Dateien: Satzmaske festlegen
	GET	Relative Dateien: Lesen von Datensätzen
	PUT	Relative Dateien: Schreiben von Datensätzen
	FAST	Schnellere Bildschirmausgabe im Mode 2
	SLOW	Normale Bildschirmausgabe im Mode 2
	DERROR	Abfangen von Diskettenfehlern
	FRAME	Synchronisation des Bildrücklaufs (Frame)
	GCHAR	Lesen von Zeichen des Bildschirms (Getchar)
	GPAPER	Grafik-Paper festlegen (Graphics Paper)
	GPEN	Grafik-Pen festlegen (Graphics Pen)
	MASK	Maske für die Grafikausgabe (Mask)
	UNMASK	Maske wieder aufheben
	M	Aufruf des Maschinensprache-Monitors

Die RSX-Befehle unter Amsdos und VDOS im Vergleich

PROGRAMM-SERVICE



Bestellungen in der Schweiz: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Tel. 042/41 56 56
Bestellungen in Österreich: Bücherzentrum Meidling, Schönbrunner Straße 261, A-1120 Wien, Tel. 0222/83 31 96,
Microcomput-ique E. Schiller, Fasangasse 21, A-1030 Wien, Tel. 0222/78 56 61,
Ueberreuter Media Handels- und Verlagsgesellschaft mbH, Alser Straße 24, A-1091 Wien, Tel. 0222/48 15 38-0
Bestellungen aus anderen Ländern bitte per Auslandspostanweisung!

Das Angebot dieser Ausgabe:

Programme für Schneider-Computer

Spiele *Öl.* Eifern Sie den Ewings nach mit Ihrem Öl-Imperium. *Feldherr.* Taktisches Spiel für zwei bis vier Personen um die Weltherrschaft. *Lunar-Lander.* Reaktion und Geschicklichkeit sind gefordert. Mit Spielfeldgenerator.
Anwendungen *Taschenrechner.* Eine Simulation für mathematisch Interessierte. *Statistik.* Statistische Daten grafisch aufbereitet.
Grafik *3D-Dreh.* Mit erstaunlich wenig Programmieraufwand betrachten Sie auf Ihrem Monitor Gegenstände in dreidimensionaler Vektordarstellung. *Background-Painter.* Hintergrundgrafiken für eigene Spiele komfortabel und mit geringstem Speicherbedarf erzeugen.
CP/M *Deutsche Tastatur für alle CPCs.* Damit Sie auf Ihrem Computer wie mit der Schreibmaschine arbeiten können. *WordStar-Verbesserung.* Unglaublich, was man aus leistungsfähigen Programmen noch mehr herausholen kann.

Diskette
Bestell-Nr. LH 86S7 SD DM 34,90*
(sFr. 29,50/öS 349,-*)

Kassette
Bestell-Nr. LH 86S7 SK DM 34,90*
(sFr. 29,50/öS 349,-*)

Programme aus früheren Ausgaben:

Happy-Computer, Ausgabe 6/86
Tron-Construction-Set: Das Spiel des Monats und Listing des Monats zugleich verspricht eine Menge Spielspaß. Auf zwei Bildschirmen bewegen zwei Spieler gleichzeitig ihre Figuren.
HiRes-Hardcopy: Das C64-Programm zum Schwerpunkt. Drucker anschließen, Programm laden und starten: Schon kann man die schönsten Hardcopies von HiRes-Bildern drucken.
Ultraload: Auch diesmal ist das sensationelle Schnelladeprogramm (Ausgabe 1/86) wieder mit auf Diskette.
Checksummer: Diese Eingabehilfe ist für Basic-Programme kaum noch wegzudenken.
MSE: Maschinensprachеprogramme sind schnell und sicher mit MSE eingegeben.
Diskette für den C64/C128
Bestell-Nr. LH 8606 CD
DM 29,90*/sFr. 24,90/öS 299,-*

Happy-Computer, Ausgabe 5/86
Commodore 64, Commodore 128
Radish-Two.
Ein Kletter- und Sammelspiel für den C64.
Ultraboot.
Ergänzung zu »Ultraload Plus«. 104 zusätzliche Blöcke auf der Diskette.
Simple Sound.
Eine kleine Soundbibliothek bietet Klänge für jede Gelegenheit.
Aus Ausgabe 4/86.
Quadrophenia.
Spiel des Monats für den Commodore 64.
Kurven.
Mathematische Kurven auf dem C128 schnell programmiert. (Läuft nicht im C64-Modus!)
Kalender.
Ein Kalender für die Jahre bis 2000.
Auto-Boot 128.
Das Programm nutzt die Fähigkeit des C128, CP/M-Programme automatisch zu booten (laden). (Nicht für C64.)
Widerstände.
Eine Utility, die Ihnen hilft, Widerstandswerte aus Farbskalen in numerische Werte umzurechnen. Aus Ausgabe 5/86.
Diskette für den C64/C128
Bestell-Nr. LH 8605 CD
DM 29,90*/sFr. 24,90/öS 299,-*

Happy-Computer, Ausgabe 4/86
Schneider CPC
D-Mon.
Daten auf Diskette Byte für Byte lesen und ändern. Fehlerhafte Dateien korrigieren und retten.
GOTO XY (nur CPC 464).
Eine mächtige RSX-Befehlserweiterung, die erlaubt, das Ziel von GOTO-GOSUB-Befehlen mit Hilfe einer Variablen zu bestimmen.
Accept.
Ein komfortabler Ersatz für den normalen INPUT-Befehl, mit dem sich jetzt die maximale Eingabe-Länge begrenzen läßt.
Turbo-Screen (nur CPC 464).
Mit dieser RSX-Erweiterung machen Sie der Bildschirmausgabe im Modus 2 Beine.
Aus Ausgabe 2/86.
Explora.
Mit diesem Prüfsummen-Generator entfällt die lästige und zeitaufwendige Fehlersuche.
Stack-Manipulation (nur CPC 464).
Basic-Programmierung mit vier RSX-Befehlen.
Aus Ausgabe 3/86.
Tool-Basic.
44 neue RSX-Befehle für Grafik-, Sprite-, Disketten- und Kassetten-Programmierung.
Achtes Bit.
Endlich Abhilfe für den Umstand, daß der Schneider CPC über die Drucker-Schnittstelle nur sieben Datenbits ausgibt.
Mord im Computer.
Das DFÜ-Spiel mit Adventure-Charakter.
Aus Ausgabe 4/86.
Best.-Nr. LH 8604 SK (Kassette)
DM 29,90*/sFr. 24,90/öS 299,-*
Best.-Nr. LH 8604 SD (Diskette)
DM 29,90*/sFr. 24,90/öS 299,-*

Happy-Computer, Ausgabe 3/86
Commodore 64/Commodore 128
Copter-Fight, Husky-Basic, Unser Sonnensystem, Wahlautomat, Softpaint
Bestell-Nr. LH 8603 CD
DM 29,90*/sFr. 24,90/öS 299,-*

Happy-Computer, Ausgabe 2/86
Commodore 64
Oval Pattern, Börse, Poster Hardcopy, Kassetten-Designer, Super-Sprite, Transbit.
Alle 6 Programme auf Diskette für den Commodore 64/128.
Bestell-Nr. LH 8602 CD
DM 29,90*/sFr. 24,90/öS 299,-*

Happy-Computer, Ausgabe 1/86
Commodore 64/Commodore 128
Bestell-Nr. LH 8601 CD
DM 29,90*/sFr. 24,90/öS 299,-*
Happy-Computer, Ausgabe 12/85
Atari 800XL/130XE/800
Bestell-Nr. LH 8512 B
DM 29,90*/sFr. 24,90/öS 299,-*
Happy-Computer, Ausgabe 12/85
Schneider CPC
Diskette für den Schneider CPC
Bestell-Nr. LH 8512 G (Kassette)
DM 29,90*/sFr. 24,90/öS 299,-*
Bestell-Nr. LH 8512 D (Diskette)
DM 34,90*/sFr. 29,50/öS 349,-*
Happy-Computer, Ausgabe 11/85
Commodore 64
Bestell-Nr. LH 8511 A
DM 29,90*/sFr. 24,90/öS 299,-*
Happy-Computer, Ausgabe 10/85
Sinclair Spectrum
Bestell-Nr. LH 8510 D
DM 19,90*/sFr. 17,-/öS 199,-*
Atari 800XL
Bestell-Nr. LH 8510 B
DM 29,90*/sFr. 24,90/öS 299,-*
Happy-Computer, Ausgabe 9/85
Commodore 64
Bestell-Nr. LH 8509 A (Diskette)
DM 29,90*/sFr. 24,90/öS 299,-*
Happy-Computer, Ausgabe 8/85
Schneider CPC 464
Bestell-Nr. LH 8508 G (Kassette)
DM 29,90*/sFr. 24,90/öS 299,-*
Happy-Computer, Ausgabe 7/85
Commodore 64
Bestell-Nr. LH 8507 A (Diskette)
DM 29,90*/sFr. 24,90/öS 299,-*
Happy-Computer, Ausgabe 6/85
Commodore 64
Bestell-Nr. LH 8506 A (Diskette)
DM 29,90*/sFr. 24,90/öS 299,-*
Happy-Computer, Ausgabe 5/85
Schneider CPC 464
Bestell-Nr. LH 8505 G (Kassette)
DM 29,90*/sFr. 24,90/öS 299,-*
Happy-Computer, Ausgabe 4/85
Commodore 64
Bestell-Nr. LH 8504 A (Diskette)
DM 29,90*/sFr. 24,90/öS 299,-*
Happy-Computer, Ausgabe 3/85
Schneider CPC 464
Bestell-Nr. LH 8503 G (Kassette)
DM 29,90*/sFr. 24,90/öS 299,-*

Happy-Sonderhefte

Sonderheft 6/86: 68000
Programme für Atari ST
Bestell-Nr. LH 86S6 D1
DM 34,90*/sFr. 29,50/öS 349,-*
Forth-Compiler für Atari ST
Bestell-Nr. LH 86S6 D2
DM 29,90*/sFr. 24,90/öS 299,-*
Programme für Apple Macintosh
Bestell-Nr. LH 86S6 D3
DM 34,90*/sFr. 29,50/öS 349,-*
Sonderheft 5/86: Programmiersprachen
Bestell-Nr. LH 86S5 SD, für Schneider
DM 34,90*/sFr. 29,50/öS 349,-*
Bestell-Nr. LH 86S5 CD, für C64
DM 29,90*/sFr. 24,90/öS 299,-*
Bestell-Nr. LH 86S5 8D, für C128
DM 29,90*/sFr. 24,90/öS 299,-*
Sonderheft 4/86: Schneider
Bestell-Nr. LH 86S4 K (Kassette)
DM 29,90*/sFr. 24,90/öS 299,-*
Bestell-Nr. LH 86S4 D (Diskette)
DM 34,90*/sFr. 29,50/öS 349,-*
Sonderheft 3/86: 68000
Bestell-Nr. LH 86S3 D (Diskette)
DM 29,90*/sFr. 24,90/öS 299,-*
Sonderheft 2/86: ATARI
Bestell-Nr. LH 86S2 D (2 Disketten)
DM 34,90*/sFr. 29,50/öS 349,-*
Sonderheft 1/86: Schneider
Bestell-Nr. LH 86S1 D (Diskette)
DM 34,90*/sFr. 29,50/öS 349,-*
Bestell-Nr. LH 86S1 K (Kassette)
DM 29,90*/sFr. 24,90/öS 299,-*
Sonderheft 2/85: Schneider
Bestell-Nr. LH 85S2 D (3"-Diskette)
DM 34,90*/sFr. 29,50/öS 349,-*
Bestell-Nr. LH 85S2 V (5¼"-Diskette)
DM 34,90*/sFr. 29,50/öS 349,-*
Bestell-Nr. LH 85S2 K (Kassette)
DM 29,90*/sFr. 24,90/öS 299,-*
Sonderheft 1/85: Spectrum
Bestell-Nr. LH 85S1 D (Kassette)
DM 19,90*/sFr. 17,-/öS 199,-*

* inkl. MwSt. Unverbindliche Preisempfehlung

Bitte verwenden Sie für Ihre Bestellung und Überweisung die eingeheftete Postgiro-Zahlkarte, oder senden Sie uns einen Verrechnungs-Scheck mit Ihrer Bestellung.
Sie erleichtern uns die Auftragsabwicklung, und dafür berechnen wir Ihnen keine Versandkosten.

Ein bombiges Kerlchen

Den Spielhallen-Renner »Bomb Jack« gibt es jetzt auch für den Schneider CPC. Neben allerfeinster Farbgrafik lockt das Programm mit fantastischem Spielwitz – ein Fest für Arkade-Freaks.

Allzeit bereit, rund um die Welt: Bomb Jack, der agile Terroristenschreck, ist wieder unterwegs. Bomben entschärfen ist sein Metier, das er trotz aller Widrigkeiten blendend beherrscht. Dieser Bursche begeistert schon seit ein paar Jahren die Besucher von Spielhallen. Das englische Softwarehaus Elite Systems hat sich die Rechte für die Heimcomputerumsetzungen gesichert. Bei der Version für die Schneider-Computer erwischte Elite Systems zum Glück die richtigen Programmierer: Die Grafik ist ein Augenschmaus und der Spielwitz steht dem Automaten-Original in nichts nach.

Die Regeln sind einfach: Sie steuern Bomb Jack, der auf jedem Bild 24 Bomben entschärfen muß. Das geschieht durch Berühren jeder einzelnen Bombe, die dann sofort verschwindet. Auf dem Bildschirm ist immer eine Bombe zu sehen, deren Zündschnur glimmt. Erwischt Bomb Jack so eine Spezialausführung, bekommt er gleich die doppelte Punktzahl – normalerweise gibt es 100 Punkte pro Knallkörper – gutgeschrieben.

Wenn es Ihnen gelingt, mindestens 20 Bomben mit brennender Zündschnur zu erwischen, winken saftige Sonder-Punkte. Bei 20 Stück gibt es 10000, bei 21 Stück 20000, bei 22 Stück 30000 und bei 23 brennenden Bomben gar 50000 Punkte extra!

Die Steuerung ist einfach. Auf Joystick-Knopfdruck hin springt Bomb Jack in die Höhe, drückt man den Joystick gleichzeitig nach oben, macht er gar einen ultra-hohen Sprung. Durch erneuten Feuerknopfdruck kann man den Sprung abbrechen und Bomb Jack sinkt wieder zu Boden. Eine besondere Technik ermöglicht es Bomb Jack, sogar waagrecht zu fliegen: Wenn man in der Sprungphase blitzschnell auf den Feuerknopf drückt, klappt die Lenkung nach links und rechts.

Doch das Schicksal schlägt wieder einmal in Form von Bösewichten zu, die über den Bildschirm kreuchen und fliegen und Bomb Jack bei Berührung ein Leben kosten. Ab und zu schwirrt aber auch ein Bällchen über dem Bildschirm, das bei Berührung Freude bringt.

Es gibt drei Sorten von Buchstaben-Bällen, die sich Bomb Jack unbedingt schnappen sollte. B-Modelle erhöhen den Bonus, den es beim Aufsammeln brennender Bomben gibt, P-Bälle erlauben einige Sekunden lang das Angreifen der Gegner, und Bälle der Marke E bringen gar ein Extraleben.

Es gibt nur fünf verschiedene Hintergrundgrafiken, doch langweilig wird es nicht. Der Aufbau der Plattformen, auf denen die Bomben liegen, ändert sich nämlich von Bild zu Bild.

»Bomb Jack« erfreute sich bei uns in der Redaktion einer ungeheuren Beliebtheit, die in abendlichen High Score-Jagden ausartete. Selbst unser Commodore-Experte ging fremd und vergnügte sich mit der Schneider-Version, wo er es auf stattliche 207000 Punkte brachte. Übertroffen wurde er nur vom absoluten Redaktions-High Score, den unser Spiele-Spezialist Heinrich »Quickfinger« hält: 250000 Punkte! Wer mehr aufs Konto bringt, möge sich doch mal mit einem Briefchen bei der Redaktion melden!

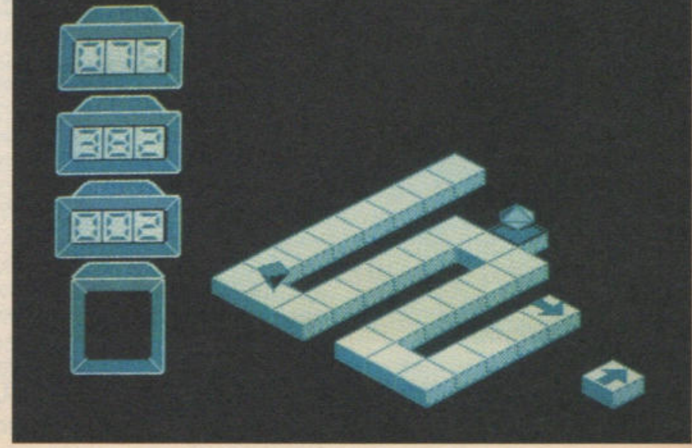
(Heinrich Lenhardt/hg)



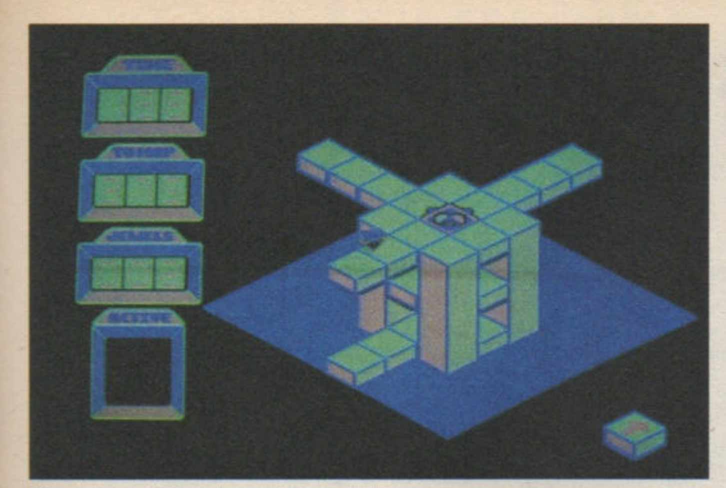
Bomb Jack muß quer durch die Welt reisen und Bomben sammeln. Hier befindet er sich gerade vor einem deutschen Traumschloß



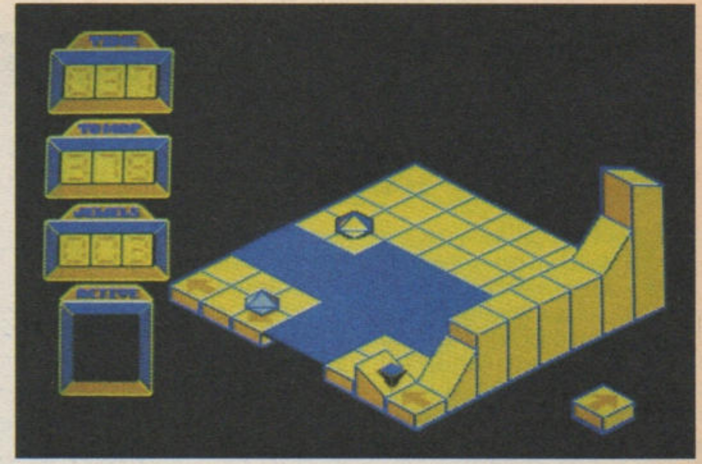
Auch Ägypten bleibt nicht vor Bombenlegern verschont



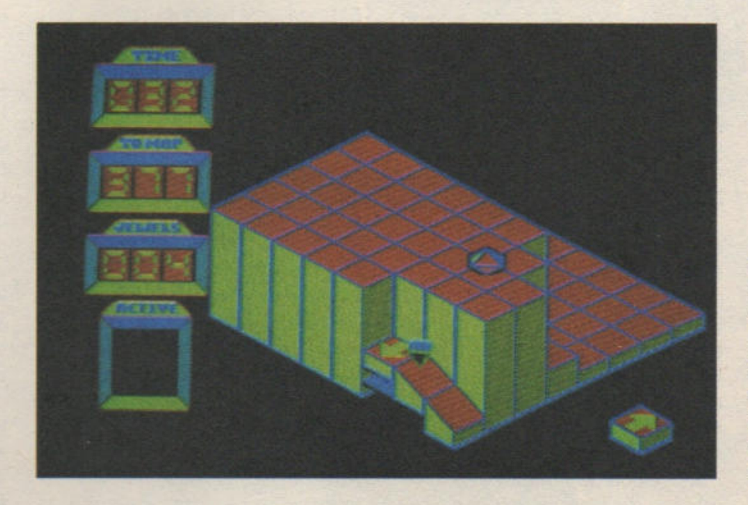
Dieses Bild zeigt schon den nächsten Spiele-Hit: Spindizzy



Hier steuert Gerald auf einen Lift zu



Mit etwas Anlauf kann Gerald das Hindernis als Sprungschanze benutzen



Der 3D-Effekt der Grafik ist verblüffend. Da man seinen Blickwinkel ändern kann, ist es möglich, die »verdeckten« Ecken eines Screens zu sehen

Welten-Bummler

Suchen Sie ein Spiel mit atemberaubender 3D-Grafik, kniffligen Puzzles und joystickgefährdender Action? Kein Problem, mit »Spindizzy« werden Sie in allen Belangen bestens bedient.

Spindizzy stellt Sie vor die Aufgabe, eine verrückte Bildschirmwelt zu erforschen und ganz nebenbei Diamanten aufzusammeln. Wenn Sie jeden der über 300 Räume besucht und alle Edelsteine eingesackt haben, ist das Spiel gelöst. Das klingt alles aber viel einfacher, als es ist: Tückische Screens mit Schikanen, Bösewichten und üblen Tricks gilt es zu meistern. Wer »Spindizzy« bewältigen will, muß ein Meister mit dem Joystick sein und hier und da auch mal kräftig nachdenken.

Die Spielfigur, die Sie steuern, hört auf den Namen Gerald. Die Bilder, durch die Gerald unterwegs ist, erinnern grafisch sehr an den Spielhallen-Kulthit »Marble Madness«. Diese Ähnlichkeit ist bestimmt kein Zufall. Bei »Spindizzy« hat es ein Programmierer erstmals geschafft, diese perspektivische Grafik im »Marble Madness«-Stil vernünftig auf einem 8-Bit-Heimcomputer zu realisieren – ein Kompliment an Paul Shirley, dem »Spindizzy« ganz allein zu verdanken ist.

Um an die Diamanten zu kommen, müssen Sie Gerald oft über halsbrecherisch enge Passagen steuern, springen und so manchen Trick anwenden. Es gibt Aufzüge, die erst aktiviert werden müssen, Trampolinfelder und Eiszonen, auf denen Gerald ganz schön ins Schleudern kommt. »Spindizzy« ist also nicht »nur« ein Geschicklichkeits-Test mit schöner Grafik, sondern ein gewitztes Spiel mit vielen unterschiedlichen Elementen. Dazu gehört auch ein gehöriger Schuß Logik. Manche Hindernisse lassen sich nur per Aufzug überwinden. Doch diese Aufzüge müssen erst per Schalter aktiviert werden. Dabei können maximal zwei Aufzüge eingeschaltet sein.

Komfortabel ist die Edelstein-Hatz auch: Durch Druck auf »M« erscheint eine Karte, »S« zeigt den aktuellen Punktestand an, »C« schaltet auf einen speziellen Modus für monochrome Monitore um, »P« hält das Spiel an und mit »I« kann man sogar Gerald's Aussehen verändern: Drei Formen stehen dabei zur Auswahl. Über die Cursortasten können Sie wählen, von welcher Seite aus Sie das Spielfeld betrachten. Da die Grafik perspektivisch erscheint, kommt man bei einigen Bildern ohne diese Hilfe gar nicht weiter.

»Spindizzy« und das oben erwähnte »Bomb Jack« sind nach unserer Meinung die derzeit besten Actionspiele für den Schneider. Sie bieten beide hervorragende Grafik, erstklassigen Spielwitz und als entscheidendes Kriterium eine langfristig hohe Motivation. Beide Programme sind auf Kassette und 3-Zoll-Diskette erhältlich und kosten je nach Datenträger zwischen 35 und 60 Mark.

(Heinrich Lenhardt/hg)

Bezugsquelle: Korona Soft, Postfach 3115, 4830 Gütersloh 1



ROM-Listings im Vergleich

Mit der Verbreitung der Schneider-Computer wächst auch die Zahl der Buchveröffentlichungen. Hier eine Auswahlhilfe für Maschinensprache-Programmierer.

Drei Dinge braucht ein Maschinensprache-Programmierer beim Schneider CPC: einen Assembler, ein Firmware-Handbuch und ein ROM-Listing. Doch gerade bei den ROM-Listings gibt es reichlich Auswahl. Welche sind besonders empfehlenswert? Wo liegen ihre Schwächen und ihre Stärken?

CPC 464 inside out

Beginnen wir mit einem der ältesten auf dem Markt, dem Buch »CPC 464 inside out«. Es ist – wie der Name schon erkennen läßt – ausschließlich auf den CPC 464 zugeschnitten. Nachfolgebände für den CPC 664 und CPC 6128 sind bis jetzt noch nicht erschienen.

Das Buch gliedert sich in unterschiedliche Kapitel. Am Anfang steht eine Beschreibung der RAM/ROM-Umschaltung und des internen Aufbaus von Basic-Programmen. Darauf folgt das ROM-Listing, das durch Verwendung farbiger Seiten auch optisch aufgeteilt wurde: Auf weißen Seiten steht das Disassembler-Listing des Betriebssystems, auf den gelben Seiten das komplette System-RAM mit den Systemvariablen und Sprungvektoren. Daran schließen sich wieder weiße Seiten an, die den Inhalt des Basic-ROMs wiedergeben.

Nachteilig ist, daß der Autor sämtliche Kommentare zu den ROM-Listings in englischer Sprache gehalten hat. Was er damit bezweckt, ist etwas unklar, da alle übrigen Texte in Deutsch sind. Sehr positiv zu vermerken ist hingegen die »stack-orientierte« Schreibweise in den Listings. Nach einem PUSH-Befehl erscheinen die folgenden Z80-Befehle eingerückt, bis durch POP der Stapelzeiger SP wieder auf den alten Stand gebracht wird. So hat der Leser stets einen Überblick darüber, wie ein ROM-Programm den Stapelspeicher verwaltet. Äußerst nützlich sind auch die Querverweise auf andere Teile des ROMs und RAMs, die jede ROM-Routine und Systemvariable, sei sie nun offiziell bekanntgegeben oder nicht, begleiten. Ein Kennbuchstabe macht ersichtlich, ob auf die Adresse schreibend oder lesend zugegriffen, ob sie als Datum oder als Unterprogramm verwendet wird. In diesem Buch, als einzigem der drei getesteten, sind auch die RAM-Routinen (!) ab der Adresse B921 hex disassembliert.

CPC 464 Intern und CPC 664/6128 Intern

Das »464 Intern« gehört ebenfalls zu den schon etwas älteren Büchern (wenn man etwa eineinhalb Jahre »alt« nennen will). Es entwickelte sich inzwischen zu einer Art »Standardwerk« für den Schneider CPC 464.

Das Buch beschreibt sehr ausführlich die Hardware, die einzelnen Chips im Computer und die Komponenten des Betriebssystems. Auch die Einbindung von RSX-Befehlen, die Manipulation der nur beim CPC 464 vorhandenen »Error- Patches«, der Aufbau von Basic- Programmzeilen und die Verwaltung des Basic-Stacks sind erläutert. Zum ROM-Listing selbst kann man eigentlich recht wenig sagen. Man vermißt allerdings die Einrückungen und Querverweise. Die Tabelle im Anhang ist nicht so vollständig wie im »inside out«. Dafür sind die Kommentare in deutscher Sprache gehalten und zum größten Teil recht ausführlich. Bei einigen Routinen des Betriebssystems hat sich das Autorenteam allerdings nicht allzuviel Mühe gegeben. So kommt der Sound-Manager besonders schlecht weg und erfährt über Seiten hinweg fast keine Kommentierung. Da informiert »inside out« eindeutig besser. Andererseits ist für den Bastler noch der Schaltplan des Computers im Anhang des Buches von Interesse. Deshalb kann man das »464 Intern« guten Gewissens empfehlen. Beim »664/6128 Intern« hat Data-Becker alle Disassembler-Listings aus dem Buch verbannt. Daher kam es zu der etwas seltsam anmutenden Situation, daß nur noch die Kommentare dazu abgedruckt sind. Im Anhang ist dann auf 15 Seiten ein in Basic geschriebener Disassembler zu finden. Diesen muß man erst abtippen und mit ihm die passenden Teile des ROMs disassemblieren, bevor man mit den Kommentaren im Buch etwas anfangen kann. Dies ist allerdings zeitaufwendig und nicht gerade der Sinn eines ROM-Listing-Buches. Außerdem setzt es beim Leser einen Drucker voraus. Ansonsten ist das »664/6128 Intern« im Textteil weitgehend identisch mit dem »464 Intern«. Nur ein Kapitel über die hard- und softwaremäßige Integration der zweiten 64-KByte-Speicherbank beim CPC 6128 wurde zusätzlich aufgenommen.

ROM-Listing CPC 464/664/6128

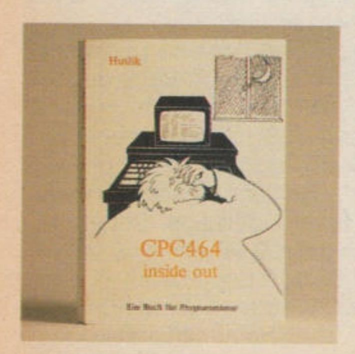
Dieses Buch ist das jüngste der vorgestellten und auch das umfangreichste. Es umfaßt immerhin fast 700 Seiten. Die beiden Autoren hatten sich vorgenommen, nicht nur ein komplettes ROM-Listing des CPC 464 vorzulegen, sondern auch auf die Unterschiede zwischen den ROMs der drei Schneider CPCs einzugehen. Eine Auflistung der ROMs von CPC 464, CPC 664 und CPC 6128 wäre natürlich viel zu umfangreich gewesen. So findet der Leser ein komplettes Disassembler- Listing des CPC 464-ROMs sowie auf weiteren 50 Seiten Listings derjenigen Betriebssystem- und Interpreter-Teile der Modelle 664 und 6128, die sich erheblich von denen des 464-ROMs unterscheiden. Eine durchaus sinnvolle Lösung.

Den Anfang des Buches bildet – wie gehabt – eine Einführung in die grundlegenden Strukturen der Hard- und Software des Schneider CPC. Beim »ROM-Listing CPC 464/664/6128« gerieten sie aber besonders ausführlich. Dinge wie Ringpuffer, LIFO- und FIFO-Prinzipien und die theoretischen Grundlagen der Arithmetikroutinen, die hier zur Sprache kommen, sind bei keinem der anderen ROM-Listings zu finden. Positiv ist außerdem zu werten, daß stets Vergleiche zwischen den verschiedenen CPC-Versionen gezogen werden, so daß der Leser sofort erfährt, inwieweit eine bestimmte ihm vorschwebende Software-Lösung überhaupt portabel ist. Die Besprechung der Speichererweiterung beim CPC 6128 ist zwar umfassender als beim »CPC 664/6128 Intern«, aber leider auch noch nicht völlig korrekt.

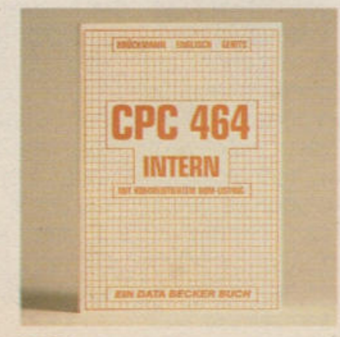
Eine wirkliche Meisterleistung haben die Autoren vollbracht, indem sie sämtliche Routinen des Betriebssystems und des Basic-Interpreters sowie des System- und Basic-RAMs nebeneinander für alle CPCs auflisten. Hier kann man nachschlagen, wenn man auf eine ROM-Routine oder Systemvariable bei einem der CPCs gestoßen ist und wissen will, welche Adresse sie bei den anderen CPCs besitzt. Programmkonversionen zwischen den verschiedenen Modellen sind dadurch wirklich kinderleicht. Ein ausführliches Stichwortverzeichnis rundet den Eindruck des Buches positiv ab.

(Elisabeth Stenzel/ja)

Info:
Winfried Huslik, »CPC 464 inside out«, S. Huslik Verlag, Augsburg 1985, ISBN 3-925159-00-2, 59 Mark.
Brückmann, Englisch, Gerits, »CPC 464 Intern«, Data-Becker, Düsseldorf 1985, ISBN 3-89011-080-0, 69 Mark.
Brückmann, Englisch, Gerits, Steigers, »CPC 664/6128 Intern«, Data-Becker, Düsseldorf 1985, ISBN 3-89011-135-1, 69 Mark.
Janneck, Mossakowski, »ROM-Listing CPC 464/664/6128«, Markt & Technik, Haar bei München 1986, ISBN 3-89090-134-4, 64 Mark.



Inside out – der Oldtimer



CPC Intern – das Standardwerk

ROM-Listing CPC – drei auf einen Schlag


SPIELE ZU GEWINNEN
Unter den Einsendern dieses Coupons werden 25 Exemplare des klassischen Arcade-Spiels AIRWOLF verlost.
Der Rechtsweg ist ausgeschlossen.
Name und Adresse:
Computer-Typ:
Name des Magazins, welchem Sie diesen Coupon entnommen haben:
Senden Sie diesen Coupon an:
PETER WEST RECORDS GmbH
Am Heerdter Hof 15
4000 Düsseldorf 11
GHOSTS'N GOBLINS
SPACE INVASION
AMSTRAD/SCHNEIDER
elite
BOMB JACK
BASIC
CONSTRUCTION
ARCADE
Der Knaller!!!!
Eine Basicerweiterung für den CPC 464 ohne RSX.
ACB erlaubt das einfache Programmieren von Arcade-Spielen.
Sehr einfach zu handhaben.
Alle Befehle werden direkt eingegeben.
Zu dem Paket gehören ein SPRITE EDITOR und ein
AUSFÜHRLICHES DEUTSCHES HANDBUCH.
Händleranfragen an:
PETER WEST RECORDS GmbH
Am Heerdter Hof 15
4000 Düsseldorf 11
Telefon 02 11/50 21 31
Telex 8 582 493 pwr oder
Telefax 02 11/5 04 86 19
Der Preis nur
44,90

Anpassungsprobleme

Das neue F1-X-Laufwerk für den Schneider CPC verspricht wahre Wunderdinge: rund 700 KByte Speicherkapazität sowohl in Basic als auch unter CP/M, wahlweiser Betrieb von VDOS und Amsdos, Zusammenarbeit mit der Speichererweiterungs-Karte von Vortex und einiges mehr.

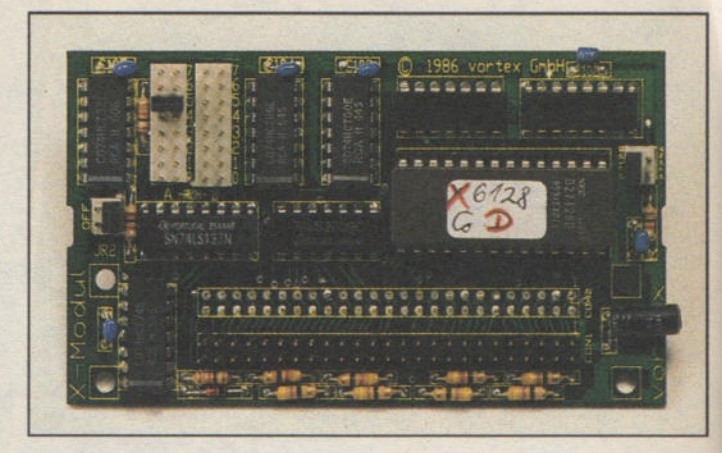


Bild 1. Der »Jumper« selektiert die ROM-Nummer

Gleichzeitig VDOS, Amsdos und CP/M verspricht das neue F1-X-Laufwerk von Vortex. Doch vor den Genuß solcher »Wunderdinge« haben die Götter den Schweiß gesetzt. Denn im mitgelieferten Handbuch ist die hard- und softwaremäßige Integration der Diskettenstation in die übrige Computer-Anlage weder ausführlich noch fehlersicher beschrieben.

Und dabei ist es ganz einfach, dieses Laufwerk mit dem CPC 464 und CPC 664 zu verbinden. Zuerst aber für die 464-Besitzer einige Hinweise zum Anschluß des Controllers, da dies auf eine reichlich ungewöhnliche Weise vor sich geht.

Sie brauchen zum Aufbau einen kleinen Kreuzschlitz-Schraubenzieher – das ist alles! Schalten Sie unbedingt als erstes den Computer und alle Peripheriegeräte ab. Dann trennen Sie den an die Konsole gesteckten Controller ab und ziehen den Stecker des Flachbandkabels vorsichtig von der 3-Zoll-Diskettenstation. Nun ist Ihr Amsdos-Controller völlig vom System getrennt und Sie wenden sich dem neuen von Vortex zu. Mit dem Schraubenzieher lösen Sie die vier Schrauben auf der Unterseite und nehmen die Abdeckung ab. Vor Ihnen liegt nun offen eine kleine Platine mit dem VDOS-Eprom und einigen anderen ICs.

Falls Sie bereits ein Erweiterungs-ROM an Ihren Computer angeschlossen haben, kann es zu Überschneidungen bei der ROM-Nummer kommen. Die Vordergrund-ROMs besitzen beim Schneider CPC 464 die Nummern 0 bis 7, beim 664 und CPC 6128 die Nummern 0 bis 15. VDOS belegt beim X-Laufwerk die Nummer 5. Und genau dasselbe macht auch der Maxam-Assembler in der ROM-Version. Das hat zur Folge, daß der Computer bereits beim Einschalten ständig aussteigt, solange beide ROMs gleichzeitig vorhanden sind. Doch Vortex hat das vorausgesehen und bietet Abhilfe. Auf der VDOS-Platine befinden sich zwei Jumperfelder (Bild 1). Das sind die weißen Rechtecke, aus denen je 16 Pins hervorstehen. Unter den Jumpern steht auf der Platine der Text »A-ROM-B«. Gefunden? Dann sehen Sie sicher auch den kleinen schwarzen Klotz, der auf den Pins der Nummer 5 steckt. Das bedeutet, daß das ROM die Nummer 5 haben wird. Um eine andere ROM-Nummer zu wählen, ziehen Sie mit den Fingern den Klotz heraus und stecken ihn im linken Jumperfeld auf zwei neue Pins – am rechten Rand stehen die jeweiligen ROM-Nummern.

Hier eine kleine Übersicht der bisher vergebenen Nummern:

7 – Amsdos-ROM
6 – BOS-EPROM für die Speichererweiterung
5 – Maxam-Assembler
0 – Basic-ROM

Es sind also noch die Nummern 1, 2, 3 und 4 frei.

Wenn Sie diese Bastelei gemacht haben, nehmen Sie wieder Ihren Amsdos-Controller zur Hand und legen den VDOS-Controller so vor sich, daß das herausstehende Flachbandkabel zu Ihnen hin zeigt. Beim Amsdos-Controller muß das Kabel links aus dem Gehäuse kommen. Nun stecken Sie vorsichtig den Amsdos-Controller auf die hervorstehende Platine im VDOS-Controller und drücken die beiden fest zusammen. Dann verschrauben Sie das Ganze wieder. Nun können Sie den VDOS-Controller, in dessen »Bauch« der Amsdos-Controller steckt, an die 3-Zoll-Diskettenstation und die Konsole anschließen. Das Vortex-Laufwerk verbinden Sie mit dem auf dem Kabel eingepreßten Stecker.

Damit sind alle hardwaremäßigen Anpassungen durchgeführt und das Computersystem ist wieder einsatzbereit. Beachten Sie bitte die Reihenfolge beim Einschalten: Zuerst kommt die 3-Zoll-Station an die Reihe, dann das Vortex-Laufwerk und zuletzt der Computer. Wenn Sie alles richtig gemacht haben, erscheint im Titelbild die folgende Meldung:

```
(C)1986 VDOS 2.0-X by vortex GmbH
```

Sollten beide Laufwerksmotoren in Betrieb sein und überhaupt nicht mehr stoppen, haben Sie höchstwahrscheinlich den Stecker von der 5¼-Zoll-Diskettenstation falsch herum auf das Flachbandkabel gesteckt. In diesem Fall müssen Sie sofort alles ausschalten und den Fehler beheben.

Kommen wir zur Software-Seite. Sie ist ungleich schwieriger zu handhaben als die Anpassung der Hardware. Unter Basic gibt es keine Probleme: »|B« schaltet auf das Zweitlaufwerk um und »|X« vertauscht die Zuordnung der Laufwerksnamen A und B. Aber unter CP/M 2.2 wird es schwierig.

Vortex empfiehlt allen Ernstes, das gepatchte CP/M auf die mitgelieferte Original-Diskette zu schreiben. Geht da etwas schief, ist die Katastrophe unvermeidlich. Vortex hat zwar für diesen Fall zugesichert, eine Ersatzdiskette zu stellen, aber ärgerlich und zeitaufwendig ist das trotzdem.

Unsere Methode ist zwar etwas umständlicher, aber dafür hundertprozentig sicher. Nehmen Sie Ihre Original-CP/M-Systemdiskette im 3-Zoll-Format und legen Sie sie ins A-Laufwerk. Dann starten Sie mit »|CPM« das Betriebssystem und übertragen mit DISCCOPY.COM den kompletten Inhalt der Seite A auf eine leere 3-Zoll-Diskette.

Lassen Sie die neue Diskette im Laufwerk und verstauen Sie die Original-Diskette wieder sicher. Mit MOVCPM.COM müssen Sie nun den Speicherplatz verkleinern:

```
A>MOVCPM 178 *
CONSTRUCTING 44k CP/M vers 2.2
READY FOR "SYSGEN" OR
"SAVE 34 CPM44.COM"
```

Jetzt schreiben Sie das neue CP/M mit SYSGEN.COM auf Ihre 3-Zoll-Diskette:

```
A>SYSGEN *
```

Lösen Sie einen Reset aus und legen Sie in das B-Laufwerk eine unformatierte $5^{1}/_{4}$-Zoll-Diskette. Mit »|FORMAT,2« formatieren Sie diese von Basic aus. Danach laden Sie CP/M, diesmal mit dem RSX-Befehl »|CPM,1«. Der Computer zeigt folgende Meldung:

```
CP/M 2.2 - Amstrad Consumer
Electronics plc
44K CP/M vers. 2.2-03/86
vortex GmbH
```

Legen Sie die mitgelieferte Vortex-Systemdiskette ins B-Laufwerk und starten Sie von dieser SYSGEN.COM:

```
A>B:
B>SYSGEN
SYSGEN 2.0    (C)1985 vortex GmbH
```

Die Frage nach der Quell-Diskette beantworten Sie mit »A« und drücken eine weitere Taste. Nun wechseln Sie im Laufwerk B die Diskette gegen die neu formatierte aus. Dann können Sie die Frage nach dem Ziellaufwerk mit »B« und einem weiteren Tastendruck beantworten. Bei »Noch eine Diskette?« tippen Sie »N« für »Nein«. Danach lösen Sie einen Reset aus und legen in B die Original-Vortex-Systemdiskette ein. Mit »|CPM,2« starten Sie wieder CP/M, diesmal vom B-Laufwerk aus. Kurz danach erscheint die Meldung: »Keine CP/M-Diskette – Wiederholen?«.

Warten Sie, bis der Laufwerksmotor abgeschaltet ist und legen Sie in das B-Laufwerk Ihre neue $5^{1}/_{4}$-Zoll-Diskette. Dann tippen Sie »J« für »Ja«, worauf der Computer den Rest von CP/M lädt. Auf dem Bildschirm erscheint:

```
44K CP/M vers. 2.2-03/86
vortex GmbH
```

Diesmal in gelber Schrift auf blauem Grund. Der Computer verwaltet jetzt die Vortex-Station als A- und die 3-Zoll-Station als B-Laufwerk. Sie können das ohne weiteres mit DIR nachprüfen. Ins $5^{1}/_{4}$-Zoll-Laufwerk kommt wieder die Vortex-Systemdiskette, von der Sie DISKTOOL.COM starten:

```
A>DISKTOOL
```

Wählen Sie aus dem erscheinenden Menü den Punkt »Komplettkopie einer Diskette erzeugen« und nennen Sie als Quelle und Ziel beide Male das A-Laufwerk. Bei »Formatieren der Ziel-Diskette?« und »Testen der Ziel-Diskette?« müssen Sie zweimal »J« eingeben. Nun kommt ein äußerst mühseliger Kopiervorgang, denn die zweimal 80 Tracks der Diskette werden von DISKTOOL in Happen zu je acht Spuren übertragen. Nicht genug damit, anscheinend auf Grund eines Programmierfehlers läuft der Diskettenmotor ungebührlich lange nach: immerhin rund 15 Sekunden. Das summiert sich erheblich, wenn man sich nicht gerade zu einer Verzweiflungstat hinreißen läßt und beschließt, die Disketten bei laufendem Motor zu wechseln.

Dafür haben Sie dann endlich eine vollständige Kopie der mitgelieferten Vortex-Diskette. Verlassen Sie jetzt wieder DISKTOOL mit ESCAPE und einer weiteren Taste. Es erscheint das wohlbekannte »Keine CP/M-Diskette – Wiederholen?«. Sie lösen einfach einen Reset aus und starten CP/M neu von der 3-Zoll-Diskette, die immer noch im Laufwerk liegt, mit dem RSX-Befehl »|CPM,1«. Schalten Sie dann auf B um und starten Sie SYSGEN.COM:

```
A>B:
B>SYSGEN
SYSGEN 2.0    (C)1985 vortex GmbH
```

Es gilt: Quell-Diskette in A, Ziel-Diskette in B, »Noch eine Diskette?« – »N« für Nein. Lösen Sie nun ein letztes Mal einen Reset aus. Mit »|CPM,2« können Sie von jetzt an das CP/M-Betriebssystem stets von der $5^{1}/_{4}$-Zoll-Diskettenstation laden. DISKTOOL erlaubt die Übertragung der Systemspuren auf beliebige weitere Disketten: Menüpunkt 2 »CP/M-Systemspuren übertragen«. (Martin Kotulla/hg)



A-D-Wandler

A-D ist die Abkürzung von »Analog-Digital«. Eine solche Signalwandlung verwendet man nicht nur zur Meßdatenerfassung. Bauen Sie einen A-D-Wandler, um beispielsweise einen Trackball abzufragen.

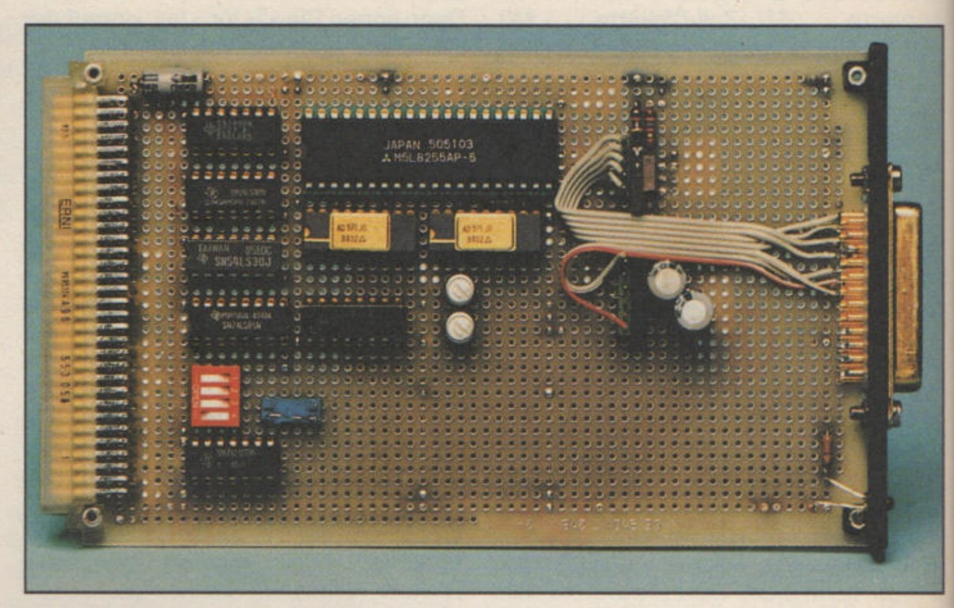


Bild 1. Der fertige A-D-Wandler für alle CPCs

In diesem Beitrag beschäftigen wir uns mit einer sehr wirkungsvollen A-D-Wandlerkarte als Trackballsteuerung. Die Vielfalt auf der Karte erklärt sich daraus, daß sie mit zwei A-D-Wandlern von je zehn Bit Breite ausgestattet ist. Dies gewährleistet eine sehr hohe Genauigkeit und Auflösung. Die A-D-Wandler setzen analoge Signale digital so um, daß sie der Computer verarbeiten kann. Die Geschwindigkeit der Wandlung ist hoch, wenn man bedenkt, daß der Wandler für zehn Bit nur 0,000015 Sekunden benötigt.

Stellen Sie sich einmal vor, Sie bewegen eine Kugel in alle Richtungen und wollen die Bewegungen mit dem Computer erfassen. Dann müssen Sie an der x- und der y-Achse die Bewegungen in analoge elektrische Signale umsetzen. In unserem Falle nehmen Sie ein Zehn-Gang-Potentiometer, das als Spannungsteiler geschaltet ist. So erhalten Sie ein analoges Signal, 'das Sie dann an die Eingänge der A-D-Wandler legen. Der A-D-Wandler wiederum setzt die Spannung, die zwischen minus 5 und plus 5 Volt liegt, am Ausgang in einen numerischen Wert zwischen 0 und 1023 um. Daraus resultiert eine Auflösung von jeweils 1024 Werten in der x- und der y-Achse. Da die abgebildete Wandler-Karte ein Muster ist, weicht sie ein wenig von der endgültigen Version ab (ein IC und zwei kleine Abgleichpotis entfallen).

Kommen wir zur Software für unsere A-D-Wandlerkarte. Das Programm (Listing) ist vollkommen interruptgesteuert und mit folgenden RSX-Befehlen in das Basic des Schneider CPC 464 eingebunden:

1. »IADON« schaltet den Trackball-Modus ein.

2. »IADOFF« schaltet ihn wieder aus.

Das Aussehen des Trackball-Cursors bestimmt »SYMBOL 255«. Über die Variablen XPOS und YPOS läßt sich die momentane Position des Trackball-Cursors ermitteln. Den Wert aus den A-D-Wandlern stellen Sie mit:

```
PRINT PEEK(&B013)+256*PEEK
(&B014)
```

für den ersten A-D-Wandler und mit

```
PRINT PEEK(&B015)+256*PEEK
(&B016)
```

für den zweiten fest.

Die erste Feuertaste läßt sich über »PEEK(&b010)« und die zweite über »PEEK(&b011)« abfragen. Wichtig ist außerdem noch die Zeile »PRINT CHR$(23)+CHR$(1)«. Sie gehört an den Anfang jedes Programms.

Nach Abtippen des Listings empfiehlt es sich, es sofort auf Kassette oder Diskette zu sichern. Man kann nun eigene Programme anfügen – beispielsweise Grafik-Programme – die mit dem Trackball zusammenarbeiten.

Kommen wir zur Praxis – dem Hardwareteil – zu unserer A-D-Wandler-Karte (Bild 1). Wir brauchen einen Lötkolben, Lötzinn, Kupfer-Lackdraht (0,2 mm Durchmesser), eine Lochraster-Leiterplatte, einige Widerstände, zwei A-D-Wandler des Typs AD 571, eine PIO 8255 und ein bißchen »digitalen Kleinkram«. Die Leiterplatte wird nach Bild 3 verdrahtet. Doch bevor Sie sich nun an das Zusammenbauen der Platine heranmachen, noch einige wichtige Tips:

- Setzen Sie möglichst alle ICs auf Sockel.
- Setzen Sie die ICs nach Lageplan ein (Bild 2).
- Die A-D-Wandler sind mit Vorsicht zu genießen (MOS-ICs).

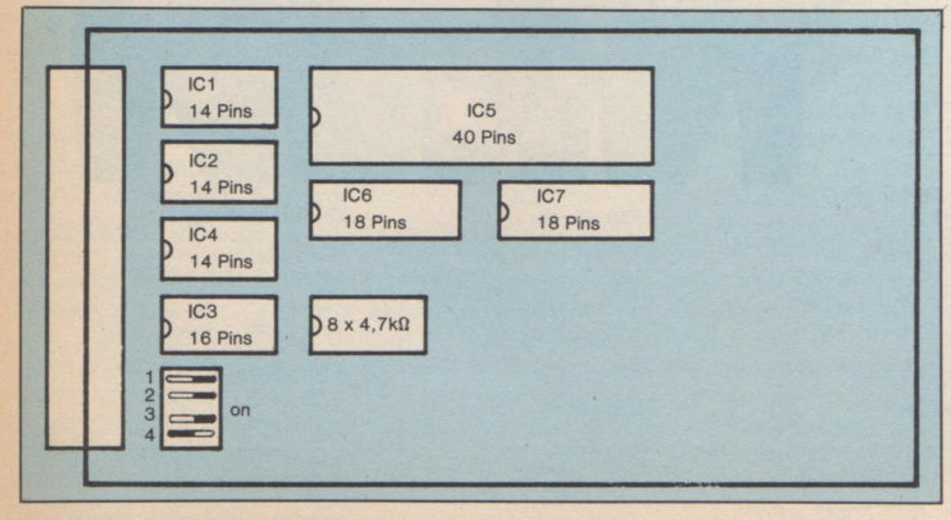


Bild 2. Lageplan der Bauteile auf der Platine

R_1 – R_7	2,7kΩ
R_8, R_9	10kΩ/10 Gang
D1, D2	
C1 – C4	10µF/16V
C5, C6	10nF

IC1	SN 74LS08
IC2	SN 74LS00
IC3	SN 74LS85
IC4	SN 74LS30
IC5	PIO 8255
IC6	AD 571
IC7	AD 571

Bild 3. Schaltplan des Wandlers

- Verwenden Sie Platinenstecker für den Anschluß an den Erweiterungs-Bus.
- Selbst angefertigte Bus-Kabel sollten auf Kurzschlüsse überprüft werden.
- Löten Sie sauber; kalte Lötstellen und Kurzschlüsse vermeiden.
- Achten Sie unbedingt auf die richtige Polung der Elkos.
- Nach Fertigstellung der Leiterplatte prüfen Sie die 5-Volt-Leitung gegen Masse auf Kurzschlüsse.
- Alle ICs müssen richtigherum in der Fassung sitzen. Sie benötigen ein Netzteil mit minus 5 und 15 Volt.

Vom Selbstbau sei abzuraten, denn 220 Volt können lebensgefährlich sein.

Die Rollkugel brauchen Sie nicht unbedingt selbst zu bauen; es gibt fertige Trackballs zu kaufen. Dann allerdings ist ein wenig Bastelei vonnöten, damit die Rollkugel mit der Steuerkarte zusammenarbeitet. Das Arbeitsprinzip eines gekauften Trackballs beruht auf der Verwendung von Reed-Relais und Magneten auf beiden Achsen. Die Mechanik ist jederzeit für unsere Zwecke brauchbar, wenn Sie anstelle der Magneten die Zehn-Gang-Potis einsetzen. Dazu bohren Sie in die Achse der Übertragungsrolle ein Loch mit etwa dem Durchmesser der Achse des Potis, und stecken ihn dort ein. Wichtig ist, daß die Achse des Potis fest in der Rolle sitzt. Jetzt bauen Sie die veränderten Achsen wieder in das Gehäuse ein. Wenn das alles soweit erledigt ist, verdrahten Sie die Potis R8 und R9 mit den beiden Germanium-Dioden D1 und D2, wie Bild 2 zeigt. Den Eingang F1 legen Sie über die Feuertaste an plus 5 Volt.

(Oliver Harms/ja)



```
10 'TRACKBALL LOADER                                [A91E]
20 SYMBOL 255,128,192,224,240,248,252,16
   ,8                                               [045E]
30 CLS                                              [EBCA]
40 PRINT CHR$(23)+CHR$(1)                           [3D2C]
50 MEMORY &8FFF                                     [6D5C]
60 DATA 21,76,90,6,4,E,0,11,0,91                    [BAC0]
70 DATA CD,EF,BC,21,70,90,11,8,0,1                  [A124]
80 DATA 1,0,CD,E9,BC,21,76,90,CD,F2                 [53E2]
90 DATA BC,21,56,90,6,1,E,0,11,6A                   [3580]
100 DATA 91,CD,EF,BC,21,50,90,11,20,0               [473A]
110 DATA 1,3,0,CD,E9,BC,21,56,90,CD                 [5CA8]
120 DATA F2,BC,0,0,0,0,0,0,0,0                      [6CEC]
130 DATA 0,0,0,0,0,0,0,0,0,0                        [52B4]
140 DATA 0,0,0,0,0,0,0,0,0,0                        [E2B6]
150 DATA 0,0,0,0,0,0,0,0,0,0                        [B2B8]
160 DATA 0,0,0,0,0,0,0,0,0,0                        [22BA]
170 DATA 0,0,0,0,0,0,0,0,0,0                        [4ABC]
180 DATA 0,0,0,0,0,0,0,0,0,0                        [0ABE]
190 DATA 0,0,0,0,0,0,0,0,0,0                        [2AC0]
200 DATA 0,0,0,0,0,0,0,0,0,0                        [B5B0]
210 DATA 0,0,0,0,0,0,0,0,0,0                        [E5B2]
220 DATA 0,0,0,0,0,0,0,0,0,0                        [4DB4]
230 DATA 0,0,0,0,0,0,0,0,0,0                        [25B6]
240 DATA 0,0,0,0,0,0,0,0,0,0                        [1DB8]
250 DATA 0,0,0,0,0,0,0,0,0,0                        [BDBA]
260 DATA 0,0,0,0,0,0,0,0,0,0                        [55BC]
270 DATA 0,0,0,0,0,0,0,0,0,0                        [DDBE]
280 DATA 0,0,0,0,0,0,0,0,0,0                        [E5C0]
290 DATA 0,0,0,0,0,0,0,0,0,0                        [95C2]
300 DATA 0,0,0,0,0,0,0,0,0,0                        [DCB2]
310 DATA 0,0,0,0,0,0,1,F3,F8,3E                     [3F74]
320 DATA 93,ED,79,1,F2,F8,3E,20,ED,79               [2CB4]
330 DATA 1,F2,F8,3E,0,ED,79,1,F3,F8                 [2BD8]
340 DATA 3E,9B,ED,79,1,F2,F8,ED,78,E6               [6706]
350 DATA 10,C2,1C,91,1,F2,F8,ED,78,E6               [8398]
360 DATA 80,32,10,B0,1,F2,F8,ED,78,E6               [7472]
370 DATA 40,32,11,B0,1,F2,F8,ED,78,E6               [036E]
380 DATA C0,32,12,B0,1,F0,F8,ED,78,5F               [038C]
390 DATA 1,F2,F8,ED,78,E6,3,57,1,F1                 [EAC6]
400 DATA F8,ED,78,6F,1,F2,F8,ED,78,E6               [B90C]
410 DATA C,1F,1F,67,ED,53,13,B0,22,15               [1654]
420 DATA B0,C9,ED,5B,13,B0,2A,15,B0,0               [A276]
430 DATA 0,1B,1B,1B,1B,1B,1B,1B,2B,2B               [D974]
440 DATA 2B,2B,2B,2B,2B,2B,2B,D5,E5,CD              [1248]
450 DATA C6,BB,D5,E5,CD,C0,BB,3E,FF,CD              [39F0]
460 DATA 19,BD,CD,FC,BB,E1,D1,CD,C0,BB              [76DA]
470 DATA 3E,FF,CD,19,BD,CD,FC,BB,E1,D1              [C1F6]
480 DATA CD,C0,BB,C9,C9,0,0,0,0,0                   [09D0]
490 DATA 0,0,21,50,90,CD,EC,BC,C9,0                 [2BA8]
500 DATA 0,0,0,0,0,0,0,0,0,0                        [E6B6]
510 DATA 0,0,0,0,0,0,0,0,0,0                        [B6B8]
520 DATA 0,0,0,0,0,0,0,0,0,0                        [26BA]
530 DATA 0,0,0,0,0,0,0,0,0,0                        [4EBC]
540 DATA 0,0,0,0,0,0,0,0,0,0                        [BEBE]
550 DATA 0,0,0,0,0,0,0,0,0,0                        [1EC0]
560 DATA 0,0,0,0,0,0,0,0,0,0                        [DEC2]
570 DATA 0,0,3E,C3,32,16,AC,21,27,92                [90AE]
580 DATA 22,17,AC,32,19,AC,21,4A,92,22              [289A]
590 DATA 1A,AC,32,7,AC,21,6B,92,22,8                [9DF8]
600 DATA AC,32,A,AC,21,7E,92,22,B,AC                [603C]
610 DATA C9,11,91,92,CD,27,E3,D0,7E,CD              [F31E]
620 DATA 7B,FF,D8,F1,F1,1A,D1,C1,F5,7               [5AF4]
630 DATA 3E,E0,38,2,3E,FF,CD,25,DF,F1               [22F8]
640 DATA CD,25,DF,C3,86,DF,79,7,30,11               [C3A4]
650 DATA 3E,E0,B9,C0,D1,C1,DD,E1,E1,7E              [2D6A]
660 DATA 23,E5,DD,E5,C5,D5,4F,21,91,92              [1A0C]
670 DATA CD,13,E3,D0,7E,23,C1,C1,C9,FE              [2054]
680 DATA C0,C0,23,7E,87,D0,C1,4F,6,0                [FD04]
690 DATA EB,21,9D,92,9,C3,BB,DD,D6,1E               [2A04]
700 DATA FE,80,D0,D6,40,D8,C1,4F,6,0                [D324]
710 DATA E5,21,A1,92,C3,B4,D0,41,44,4F              [42BE]
720 DATA CE,80,41,44,4F,46,C6,81,0,A1               [3A48]
730 DATA 92,A7,92,E5,CD,0,90,E1,C9,E5               [2BA8]
740 DATA CD,B0,91,E1,C9,0,0,0,0,0                   [468C]
750 DATA 0,0,0,0,0,0,0,0,0,0                        [58C4]
760 DATA 0,0,0,0,0,0,0,0,0,0                        [C0C6]
770 DATA 0,0,0,0,0,0,0,0,0,0                        [20C8]
780 DATA 0.0.0.0.0.0.0.0.0.0                        [98CA]
790 DATA 0,0,0,0,0,0,0,0,0,0                        [00CC]
800 DATA 0,0,0,0,0,0,0,0,0,0                        [CBBC]
810 DATA 0,0,0,0,0,0,0,0,0,0                        [F3BE]
820 DATA 0,0,0,0,0,0,0,0,0,0                        [33C0]
830 FOR i= 36864 TO 37632:READ a$:POKE i
    ,VAL("&"+a$):NEXT i                             [04C8]
840 CALL &9200                                      [D1D2]
850 'LADE BASIC PROGRAMM                            [4202]
1000 RUN"!"                                         [C376]
```

Listing. Steuerprogramm als Basic-Lader

Des Schneiders Tastatur

Haben Sie sich auch schon darüber geärgert, daß auf Ihrem Schneider die Tasten weder nach DIN angeordnet, noch Umlaute zu finden sind? Mit einer kleinen Bastelei und einem Programm können Sie das ändern. Die Tastatur kann auch unter CP/M neu belegt werden.

Wer von einer Schreibmaschine auf einen Schneider-Computer umsteigt, der verwünscht schnell die eingebaute amerikanische Tastatur. Dabei ist Abhilfe sehr einfach. Nur ein paar Basic-Befehle passen den Zeichensatz und die Tastaturbelegung an die deutsche Sprache an. Wer dann perfekt Schreibmaschine im Zehnfingersystem schreiben kann, ohne auf die Tasten zu blicken, der braucht keinen weiteren Umbau an seinem Computer vorzunehmen. Wenn Sie aber nach dem System Adler (»kreisen und zustoßen«) schreiben, dann müssen Sie auch die Beschriftung an die neue Anordung anpassen.

Insgesamt 16 Tasten sind neu zu beschriften. Austauschtasten gibt es leider nicht zu kaufen, so daß nur Eigenbau hilft. Die Kosten liegen bei unserer Bastelei unter zehn Mark. Etwas handwerkliches Geschick ist allerdings notwendig.

Als erstes müssen wir die zu verändernden Tasten abbauen. Beim CPC 464 und beim 664 braucht man dazu nur einen Schraubenzieher unter den Fuß der Taste zu schieben, diese herauszuhebeln, und schon hat man sie in der Hand. Allerdings ist etwas Vorsicht geboten, denn schnell ist das Gehäuse des Computers zerkratzt oder beschädigt. Beim CPC 6128 muß man vor dem Ausheben mit dem Schraubenzieher mit einem gebogenen Stahldraht eine der beiden Halteklauen zurückzudrücken. An der anderen Seite kann man dann die Taste herausheben. Bild 1 zeigt die beiden Methoden.

Wer eine schöne Handschrift hat, der kann jetzt seine Tasten neu beschriften. Wer nicht, der sollte zu Abreibebuchstaben aus dem Handel greifen. Es gibt verschiedene Typen von mehreren Firmen. Für zirka drei Mark pro Bogen bietet sie jedes Schreibwarengeschäft an. Mit etwas Fantasie kann man eventuell fehlende Zeichen aus anderen Symbolen herstellen.

Die Abreibebuchstaben kommen, nachdem man die aufgedruckten Symbole mit etwas Schleifpapier entfernt hat, (vorsichtig arbeiten, damit die Pla-

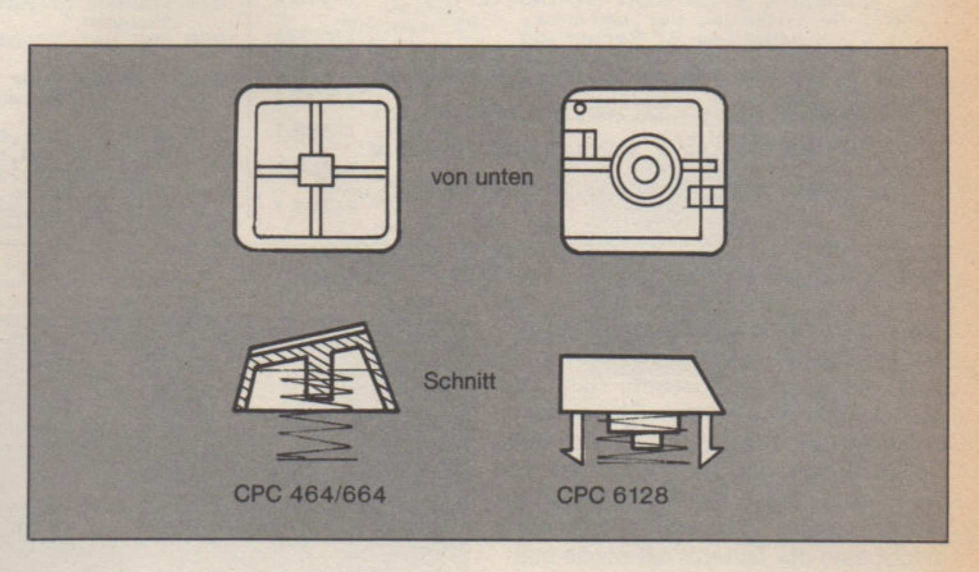


Bild 1. Die Tasten des CPC 464 und 664 unterscheiden sich von denen des CPC 6128 durch zwei Haltebügel

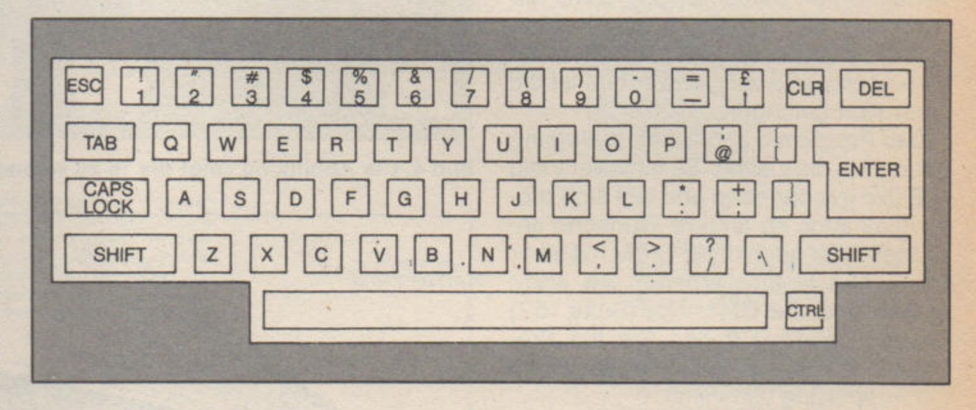


Bild 2. Die amerikanische Tastaturbelegung

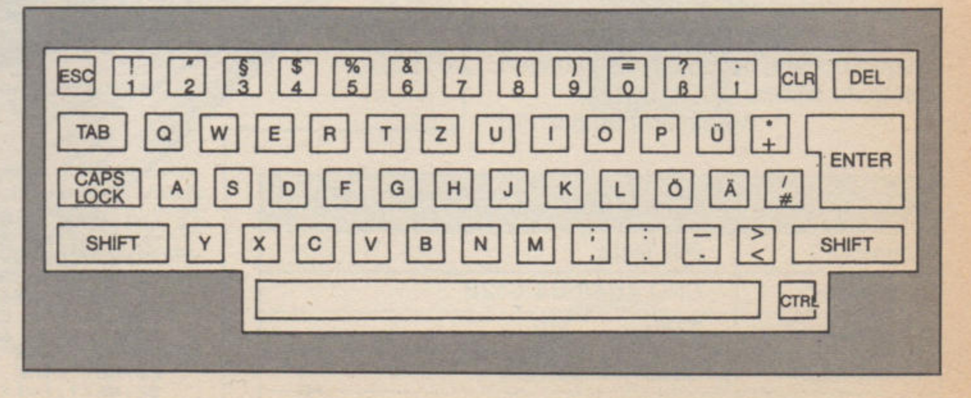


Bild 3. Die deutsche Tastaturbelegung

```
10   ***** Deutsche DIN-Tastatur *****
         ***** (c) Michael Strasser   ****
   *        *****     Rottalstr. 5        *
   ****        *****      8000 Muenchen 80
    *****                                   [2300]
20 MODE 1                                   [E5F0]
30 LOCATE 10,1:PEN 1:PRINT"Deutsche DIN-
   Tastatur"                                [FE8E]
40 SYMBOL AFTER 32                          [EFE2]
50 o$="# ´ _ = # ! { * + } < > ? `":u$="
   3 7 0 - ^ @ [ : ; ] , . / \"             [B31E]
60 ' Anzeigen der normalen Tastatur-
        belegung                            [28C6]
70 PEN 1:LOCATE 9,5:PRINT o$:LOCATE 9,12
   :PRINT u$:PEN 3:LOCATE 1,5:PRINT"ASCI
   I":LOCATE 1,12:PRINT"ASCII"              [D412]
80 PEN 2:LOCATE 9,6:PRINT STRING$(27,"="
   ):LOCATE 9,13:PRINT STRING$(27,"=")      [7FAE]
90 ' Definierung der Umlaute,
        scharfes S, Paragraph               [D5D0]
100 SYMBOL 64,60,96,60,102,60,6,60,0        [2FE2]
110 SYMBOL 91,198,16,56,108,198,254,198,
    0                                       [A21C]
120 SYMBOL 92,102,56,108,198,198,108,56,
    0                                       [C906]
130 SYMBOL 93,102,0,102,102,102,102,60,0
                                            [8B36]
140 SYMBOL 123,108,0,120,12,124,204,118,
    0                                       [C0AE]
150 SYMBOL 124,102,0,60,102,102,102,60,0
                                            [5A36]
160 SYMBOL 125,102,0,102,102,102,102,59,
    0                                       [7CA4]
170 SYMBOL 126,120,198,198,252,198,198,2
    48,192                                  [892A]
180 RESTORE 270                             [A92C]
190 ' Tasten neu definieren                 [8122]
200 FOR n=1 TO 17:READ a,b,c:KEY DEF a,1
    ,b,c:NEXT                               [15B4]
210 o$="@ / = ? ` ] * \ [ ´ ; : _ >":u$=
    "3 7 0 ~ ^ } + ! { # , . - <"           [C730]
220 ' Anzeigen der neuen Tastatur-
        belegung                            [F160]
230 PEN 1:LOCATE 9,7:PRINT o$:LOCATE 9,1
    4:PRINT u$:PEN 3:LOCATE 1,7:PRINT"De
    utsch":LOCATE 1,14:PRINT"Deutsch"       [671A]
240 LOCATE 13,20:PEN 1:PRINT"Y und Z ver
    tauscht"                                [CE38]
250 FOR n=1 TO 3000:NEXT:LOCATE 1,24:END
                                            [293E]
260 ' Datas fuer Tastaturumdefinierung
        (Tastennummer, Zchn 1, Zchn 2)
                                            [4DBA]
270 DATA 71,121,89,39,44,59,31,46,58,30,
    45,95,22,60,62,29,124,92,28,123,91,1
    9,35,39,43,122,90,26,125,93,17,43,42
    ,57,51,64,56,52,36,41,55,47,32,48,61
    .25,126,63,24,94.96                     [2E20]
```

Listing. Tastaturbelegung

stiktasten nicht zu heiß werden) auf die Tasten. Bei der Tastatur des 664 muß man allerdings aufpassen, um die Tasten aus verschiedenen Reihen nicht durcheinanderzubringen. Denn anders als beim 464 und beim 6128 ist die Form der Tasten der verschiedenen Reihen nicht identisch. Abriebfest macht man die Tastatur mit etwas mattem Klarlack.

Falls jemand seine Handschrift bevorzugt, muß er die Symbole mit einem wasserfesten Stift auftragen. Denn nur dann kann die Schrift unter dem Klarlack nicht verlaufen. Nun die Tasten einfach noch einsetzen, und schon ist die Operation gelungen.

Das Programm zur Tastaturbelegung finden Sie als Listing. Es ist vollständig in Basic geschrieben und läßt sich problemlos auch an andere Wünsche anpassen. Wer auch unter CP/M mit einer deutschen Tastatur arbeiten will, sollte einmal unter den CP/M-Tips (Seite 152) nachsehen. Dort finden Sie ein Programm, das in diesem Betriebssystem die Anpassung vornimmt.

Da man sicher auch in Zukunft manchmal mit der Originaltastatur arbeitet, sollte man die alten Symbole an der Vorderseite der Tasten anbringen. Denn sonst steht Ihnen später einige Sucherei bevor, um die richtigen Tasten zu finden.

(Hans-Otto Schulz/hg)

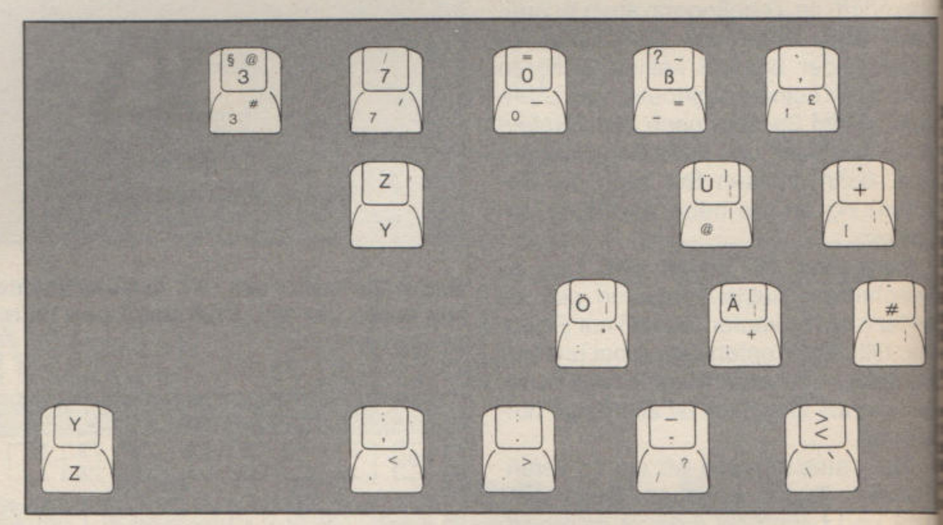
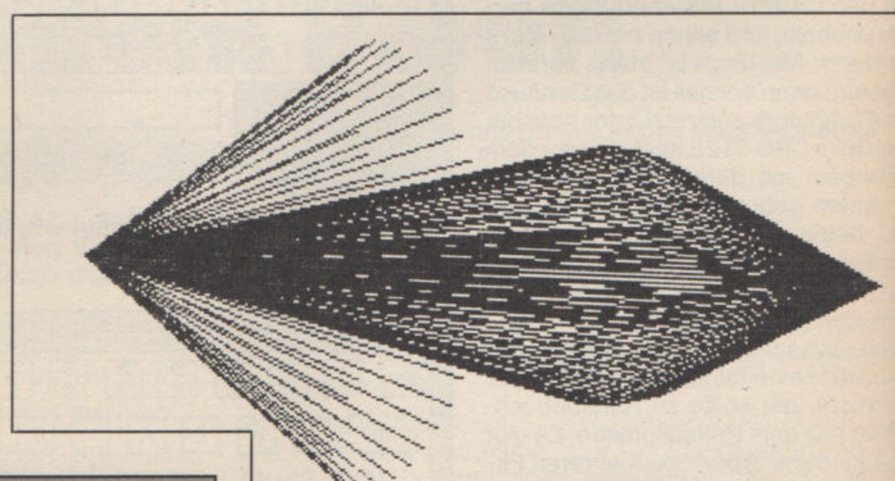


Bild 4. Die Abbildung zeigt die 16 zu verändernden Tasten

Programm-Steckbrief	
Name:	DIN-Taste
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette
Besonderes:	Hardware-Bastelei



```
10 i=1:MODE 1:BORDER 0:INK 0,0:INK 1,1:I
   NK 2,3:INK 3,6:FOR w=0 TO PI STEP PI/
   75:q1=COS(w*3)*200:w1=w+20:q2=COS(w1)
   *200:PLOT COS(w)*q1+320,SIN(w)*q1+200
   ,1:DRAW COS(w*PI)+10,SIN(w1)*COS(q2)+
   200,1:DRAW COS(w1)*q2+320,SIN(w1)*q2+
   200,2:DRAW 635,200,3:NEXT w           [3614]
20 INK 1,i:INK 2,6:INK 3,3:FOR wa=1 TO 2
   00:NEXT:INK 2,3:INK 3,6:FOR wa=1 TO 2
   00:NEXT:i=i+1:IF i=30 THEN i=1        [B09A]
30 GOTO 20                               [017C]
```

Spitzen-Software für Schneider-Computer und Commodore 128 PC

BRANDNEU
Jetzt auch für den Schneider Joyce

WordStar 3.0 mit MailMerge Der Bestseller unter den Textverarbeitungsprogrammen für PCs bietet Ihnen bildschirmorientierte Formatierung, deutschen Zeichensatz und DIN-Tastatur sowie integrierte Hilfstexte. Mit MailMerge können Sie Serienbriefe mit persönlicher Anrede an eine beliebige Anzahl von Adressen schreiben und auch die Adreßaufkleber drucken.
WordStar/MailMerge für den Schneider CPC 464*, CPC 664*
Bestell-Nr. MS 101 (3″-Diskette)
Bestell-Nr. MS 102 (5¼″-Diskette im VORTEX-Format)
WordStar/MailMerge für den Schneider CPC 6128
Bestell-Nr. MS 104 (3″-Diskette)
WordStar/MailMerge für den Schneider Joyce PCW 8256
Best.-Nr. MS 105 (3″-Diskette)
Hardware-Anforderungen: Schneider CPC 464*, CPC 664*, CPC 6128 oder Joyce, beliebiger Drucker mit Centronics-Schnittstelle
* Der Standard-Speicherplatz beim CPC 464/664 erlaubt ohne Speichererweiterung Blockverschiebe-Operationen nur bedingt und Simultan-Drucken gar nicht.
WordStar/MailMerge für den Commodore 128 PC
Bestell-Nr. MS 103 (5¼″-Diskette)
Hardware-Anforderungen: Commodore 128 PC, Diskettenlaufwerk, 80-Zeichen-Monitor, beliebiger Commodore-Drucker oder ein Drucker mit Centronics-Schnittstelle

Markt&Technik
Schneider CPC-Software
MicroPro
WordStar 3.0
mit MailMerge für den Schneider CPC 464/664
3" Schneider-Format

Und dazu die weiterführende Literatur:

WordStar für den Schneider CPC
Best.-Nr. MT 779, ISBN 3-89090-180-8
WordStar für den Commodore 128 PC
Best.-Nr. MT 780, ISBN 3-89090-181-6



dBASE II, Version 2.41 dBASE II, das meistverkaufte Programm unter den Datenbanksystemen, eröffnet Ihnen optimale Möglichkeiten der Daten- u. Dateihandhabung. Einfach u. schnell können Datenstrukturen definiert, benutzt und geändert werden. Der Datenzugriff erfolgt sequentiell oder nach frei wählbaren Kriterien, die integrierte Kommandosprache ermöglicht den Aufbau kompletter Anwendungen wie Finanzbuchhaltung, Lagerverwaltung, Betriebsabrechnung usw.
dBASE II für den Schneider CPC 464*, CPC 664*
Bestell-Nr. MS 301 (3″-Diskette)
Bestell-Nr. MS 302 (5¼″-Diskette im VORTEX-Format)
dBASE II für den Schneider CPC 6128
Bestell-Nr. MS 304 (3″-Diskette)
dBASE II für den Schneider Joyce PCW 8256
Best.-Nr. MS 305 (3″-Diskette)
Hardware-Anforderungen: Schneider CPC 464*, CPC 664*, CPC 6128 oder Joyce, beliebiger Drucker mit Centronics-Schnittstelle
* dBASE II für den Schneider CPC 464/664 ist lauffähig mit der VORTEX-Speichererweiterung auf 128 KByte. Diese erhalten Sie direkt bei der Firma VORTEX oder bei Ihrem Computerhändler.
dBASE II für den Commodore 128 PC
Bestell-Nr. MS 303 (5¼″-Diskette)
Hardware-Anforderungen: Commodore 128 PC, Diskettenlaufwerk, 80-Zeichen-Monitor, beliebiger Commodore-Drucker oder ein Drucker mit Centronics-Schnittstelle

Markt&Technik
Schneider CPC-Software
dBASE II™
ASHTON-TATE
für den Schneider CPC 6128
3" Schneider-Format

dBASE II für den Schneider CPC
Best.-Nr. MT 837, ISBN 3-89090-188-3
dBASE II für den Commodore 128 PC
Best.-Nr. MT 838, ISBN 3-89090-189-1



MULTIPLAN, Version 1.06 Wenn Sie die zeitraubende manuelle Verwaltung tabellarischer Aufstellungen mit Bleistift, Radiergummi und Rechenmaschine satt haben, dann ist MULTIPLAN, das System zur Bearbeitung »elektronischer Datenblätter«, genau das richtige für Sie! Das benutzerfreundliche und leistungsfähige Tabellenkalkulationsprogramm kann bei allen Analyse- und Planungsberechnungen eingesetzt werden wie z.B. Budgetplanungen, Produktkalkulationen, Personalkosten usw. Spezielle Formatierungs-, Aufbereitungs- und Druckanweisungen ermöglichen außerdem optimal aufbereitete Präsentationsunterlagen!
MULTIPLAN für den Schneider CPC 464*, CPC 664*
Bestell-Nr. MS 201 (3″-Diskette)
Bestell-Nr. MS 202 (5¼″-Diskette im VORTEX-Format)
MULTIPLAN für den Schneider CPC 6128
Bestell-Nr. MS 204 (3″-Diskette)
MULTIPLAN für den Schneider Joyce PCW 8256
Best.-Nr. MS 205 (″-Diskette)
Hardware-Anforderungen: Schneider CPC 464*, CPC 664*, CPC 6128 oder Joyce, beliebiger Drucker mit Centronics-Schnittstelle
* MULTIPLAN für den Schneider CPC 464/664 ist lauffähig mit der VORTEX-Speichererweiterung auf 128 KByte.
MULTIPLAN für den Commodore 128 PC
Bestell-Nr. MS 203 (5¼″-Diskette)
Hardware-Anforderungen: Commodore 128 PC, Diskettenlaufwerk, 80-Zeichen-Monitor, beliebiger Commodore-Drucker oder ein Drucker mit Centronics-Schnittstelle

Markt&Technik
128er-Software



für den Commodore 128 PC
5¼"-Diskette im Floppy 1541-Format

MULTIPLAN für den Schneider CPC
Best.-Nr. MT 835, ISBN 3-89090-186-7
MULTIPLAN für den Commodore 128 PC
Best.-Nr. MT 836, ISBN 3-89090-187-5

Jedes Buch kostet DM 49,-
(sFr. 45,10/öS 382,20).
Erhältlich bei Ihrem Buchhändler.



Sie erhalten jedes **WordStar-**, **dBASE II-** und **MULTIPLAN-Programm** für Ihren Schneider-Computer oder Commodore 128 PC fertig angepaßt (Bildschirmsteuerung). **Jeweils Originalprodukte!** Jedes Programmpaket enthält außerdem ein ausführliches Handbuch mit kompakter Befehlsübersicht. Die VORTEX-Speichererweiterung für den Schneider CPC 464 erhalten Sie direkt bei der Firma VORTEX oder bei Ihrem Computerhändler.

Diese Markt & Technik-Softwareprodukte erhalten Sie in den Computer-Abteilungen der Kaufhäuser, bei Ihrem Computerhändler oder im Buchhandel.

Jedes Programm kostet DM 199,-*

(sFr. 178,-/öS 1890,-*)

*** inkl. MwSt. Unverbindliche Preisempfehlung**

Wenn Sie direkt beim Verlag bestellen wollen: gegen Vorauskasse durch Verrechnungsscheck oder mit der eingehefteten Zahlkarte.

Bestellungen im Ausland bitte an untenstehende Adressen.
Schweiz: Markt & Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, ☎ 042/41 56 56
Österreich: Überreuter Media, Handels- und Verlagsges. mbH, Alser-Str. 24, A-1091 Wien, ☎ 0222/48 15 38-0



Unternehmensbereich Buchverlag
Hans-Pinsel-Straße 2, 8013 Haar bei München

Wer kann, der kann!

Wenn viele stolze Besitzer eines Joyce nicht wissen, daß ihr Computer weit mehr bietet als eine komfortable Textverarbeitung, liegt das an der Werbung. Er kann aber mehr!



Nachdem Schneider den Preis für den Joyce drastisch senkte und obendrein noch ein besser ausgestattetes Schwestermodell anbot, erfreut sich dieser Computer der 8-Bit-Generation wachsender Beliebtheit. Verfolgt man jedoch die technische Entwicklung auf dem Computermarkt, erkennt man schnell, daß der Trend eindeutig in Richtung der 16- und 32-Bit-Prozessoren weist (beispielsweise Atari ST und Commodore Amiga). Was macht den Joyce nun trotzdem noch so attraktiv?

Zum einen wird er als komplettes System, bestehend aus Computer, monochromem Monitor, Drucker, Diskettenlaufwerk und einem umfangreichen Software-Paket geliefert. Dazu kommt der für eine derartige Konfiguration äußerst niedrige Preis.

Betrachtet man die Hardware, setzt Schneider Maßstäbe, was Design und Platzbedarf betrifft. Drucker und Tastatur belegen auf einem Schreibtisch ungefähr den Platz einer elektrischen Schreibmaschine. Der Monitor stellt praktisch das zentrale Gehirn der Anlage dar, in dem alles zusammenläuft. Äußerst positiv fällt auf, daß sämtliche Geräte vom Monitor mit Strom versorgt werden; das umgeht den sonst obligatorischen Kabelsalat und die Mehrfachsteckdose.

Kommunikation zwischen verschiedenen Computern ist ein Thema, das aus der heutigen Computerszene nicht mehr wegzudenken ist. So findet man an der Rückseite des Monitors einen Anschluß für ein Interface mit einem seriellen (RS232) und einem parallelen (Centronics) Ausgang. Mit einem Modem oder Akustikkoppler steht Ihnen die Welt der Mailboxen und Datenfernübertragung offen. So lassen sich Programme und Daten beliebig transferieren.

Das eingebaute Diskettenlaufwerk besitzt eine Speicherkapazität von rund 170 KByte. Die Diskette hat also zirka 340 KByte Speicherplatz, denn die Rückseite ist ebenfalls verwendbar. In der Plus-Version (PCW 8512) enthält der Joyce ein zweites Laufwerk mit einer Kapazität von 720 KByte. Für wirklich sinnvolle (weil umfangreiche) Datenbanken ist diese Ausstattung sogar unerläßlich. Zusätzlich ist der Hauptspeicher auf 512 KByte (deshalb die Zahlenkombination im Namen) erweitert. Der PCW 8256 besitzt demzufolge einen Arbeitsspeicher mit 256 KByte. Der Unterschied in der Speicherkapazität wirkt sich vor allem auf das Fassungsvermögen der RAM-Floppy aus, das beim PCW 8256 »nur« 112 KByte umfaßt, beim PCW 8512 jedoch 368 KByte. Diesen Speicherbereich spricht der Benutzer wie ein zusätzliches Laufwerk an. Der Vorteil liegt darin, daß ein Zugriff auf die Daten keine mechanischen Operationen erforderlich macht. Es findet lediglich eine interne Verschiebung der Daten statt, was einen Bruchteil der sonst benötigten Zeit beansprucht. Nervende Wartezeiten beim Nachladen wichtiger Programmteile oder Daten entfallen also.

Auch der Drucker braucht einen Vergleich mit der Konkurrenz nicht zu scheuen. Er verfügt über eine reichhaltige Ausstattung und läßt sich in seinen Funktionen völlig über Locoscript steuern.

Die mitgelieferten Disketten enthalten neben den Programmiersprachen DR Logo und Mallard-Basic sowie vielen Dienstprogrammen vor allen Dingen ein leistungsfähiges Programm zur Textverarbeitung. »Locoscript« bietet reichhaltige Funktionen für nahezu jeden erdenklichen Anwendungsfall und ist zudem auch sehr bedienerfreundlich. Unter Verwendung von Windows (Einblendungen in den Bildschirm, die Anweisungen oder Auswahlmenüs enthalten) steht der Benutzer in ständigem Kontakt mit dem Computer. Dadurch verkürzt sich die Einarbeitungsphase enorm; man kann praktisch sofort mit dem Programm arbeiten, ohne vorher die gesamte, sehr umfangreiche Bedienungsanleitung studiert zu haben. Die selbstverständlich deutsche Schreibmaschinentastatur besitzt 82 Tasten und erfreut so sicherlich jedes Sekretärinnenherz, denn die zusätzlichen, von »Locoscript« belegten Tasten, tragen ihren Teil zur einfachen Handhabung bei. Wem jedoch immer noch der Umstieg auf eine solche »Höllenmaschine« schwerfällt, für den simuliert Joyce einfach eine Schreibmaschine mit seinem Direktdruck-Modus.

Folgende Aufzählung, die keineswegs alle Funktionen umfaßt, zeigt die Vielseitigkeit von »Locoscript«: Unter-

streichen, Fettdruck, verschiedene Zeichendichten und Schriftarten, Proportionalschrift, Seitenzählung, Hoch- und Tiefstellen, Verwendung von Sonderzeichen, Änderungen des Layouts, Mischen mehrerer Textbausteine und, und, und...

Aber auch andere Textsysteme sind bereits verfügbar. So können beispielsweise Wordstar-Liebhaber ihrem vertrauten Programm treu bleiben. Gerade hier stellt sich jedoch die Frage, inwiefern die Investition für eine zusätzliche Textverarbeitung sinnvoll ist, nachdem mit »Locoscript« eine leistungsfähige Textverarbeitung zum Nulltarif zur Verfügung steht.

Als Erweiterung der Einsatzgebiete im Büro bietet sich als nächstes die große Gruppe der Datenverwaltungs-Programme an, denn in welchem Betrieb fallen keine Adressen, Kundendaten und ähnliches an? Meist schlummern diese in verstaubten Karteikästen vor sich hin und wehe, man benötigt auf die Schnelle bestimmte Informationen. Wie erholsam ist da der Umgang mit einer Datenbank im Computer. Es eröffnet ganz neue Perspektiven, wenn man auf Tastendruck eine Inventur durchführen oder mit minimalem Arbeitsaufwand ein Werberundschreiben an alle Kunden schicken kann. Die Auswahl ist auf diesem Gebiet besonders reichhaltig: So gibt es neben speziellen Problemlösungen (Adressen-/Lagerverwaltung etc.) auch universelle Datenbank-Programme wie beispielsweise »dBase«, die der Anwender an verschiedenste Zwecke anpassen kann.

Tabellenkalkulations-Programme wie »Multiplan« sind recht nahe Verwandte der Datenverwaltungen. Der Hauptunterschied liegt in der tabellarischen und damit sehr übersichtlichen Darstellung der Daten und in der Fähigkeit, beliebige Zeilen und Spalten der Tabelle rechnerisch miteinander zu verknüpfen. So ist zum Beispiel eine Artikel-Preisliste in Windeseile bei Änderung der Kalkulation komplett neu berechnet.

Wem es auf die Automatisierung des gesamten Geschäftsbetriebes ankommt, ist mit kompletten Programmpaketen gut bedient. Solche Busineß-Packs enthalten dann mehrere der genannten Einzelanwendungen für bestimmte Einsatzgebiete.

Zur Verdeutlichung erfaßter Daten eignen sich Programme für Geschäfts-Grafiken. Der geplagte Manager, der täglich aufs neue über den aktuellen Stand seines Unternehmens informiert sein muß, erhält mit Programmen wie »DR-Graph« schnellsten Überblick in Form anschaulicher Grafiken. Torten-, Balken- und Liniengrafiken sowie Histogramme auf Bildschirm oder Papier sind eben erheblich einprägsamer als endlose Zahlenkolonnen.

Sie sehen also: Die zur Zeit angebotene Software reicht eigentlich aus, um alle täglich anfallenden Arbeiten im Büro zeitsparend zu erleichtern.

Falls Sie nun denken, mit diesen Verwaltungsaufgaben wäre Ihr Joyce bereits ausgelastet, liegen Sie mit Ihrer Einschätzung weit daneben. Entsprechende Software (wie zum Beispiel »DR-Draw« und »Mica«) ermöglicht sogar technische Konstruktionszeichnungen und beliebige andere Grafiken.

Auch Benutzern, die ihre Problemlösungen selbst programmieren wollen, sind genügend Hilfsmittel an die Hand gegeben. Da ist zunächst einmal das bereits erwähnte, mitgelieferte Mallard-Basic. Es ist zum – auf CP/M-Computern weit verbreiteten – Microsoft-Basic voll aufwärtskompatibel, bietet aber viele mächtige Erweiterungen. So sind dem Programmierer mit dem relativen Dateizugriff über »Jetsam« alle Möglichkeiten geboten, eigene Datei-Programme unter Basic zu entwickeln. Schlüsselwörter dienen als Suchkriterien und gestalten eine Datenbank erst richtig komfortabel. Der Benutzer ist nun nicht

mehr auf fertige Datenbanksysteme aus dem Handel angewiesen.

Das ebenfalls im Preis enthaltene »DR Logo« ist nicht nur für Kinder geeignet, sondern kann vielmehr jedem lernwilligen Programmier-Anfänger den Einstieg erleichtern. Professionelle Programmieraufgaben lassen sich mit problemorientierten Sprachen besser umsetzen. Und so ist es kein Wunder, daß entsprechende Compiler für Ihren Joyce auf dem Markt sind. Für kaufmännische Anwendungen steht Cobol zur Verfügung, während Ingenieure und Techniker ihre Probleme eher mit Fortran lösen. Auch Pascal-Programmierer kommen mit dem entsprechenden Compiler auf ihre Kosten. Die Vorteile der strukturierten Programmierung unter Pascal veranlaßt immer mehr Programmierer, sich mit dieser Sprache auseinanderzusetzen. Auch eingeschworenen Basic-Freaks gibt der C-Basic-Compiler die Chance zum Aufstieg. Die in C-Basic verfaßten Programme laufen dank der Compilierung rund zehnmal schneller als interpretierte Programme unter Mallard-Basic.

Ein wunder Punkt des Joyce ist seine eingeschränkte Eignung als Videospiel. Vielen Einsteigern gibt gerade der Spieltrieb den Anstoß zur Auseinandersetzung mit der Computertechnik.

Anders als bei der Serie der Schneider-CPCs ist die Spielepalette recht mager. Eins der wenigen Angebote ist ein Schachspiel mit vorzüglicher dreidimensionaler Grafik. Auch Abenteurer finden ein Betätigungsfeld mit Grafik-Adventures wie »Heroes of Karn«. Konstruktive Arbeitspausen verkürzt auf angenehme Art eine Partie Bridge gegen den Joyce oder ein kleines Ping-Pong-Match.

Sicher wird sich auf dem Software-Sektor in nächster Zukunft noch vieles für den Joyce tun, so daß sowohl im Bereich der Anwendersoftware, wie auch auf dem Spielesektor noch einiges zu erwarten ist.

Von der einfachen Programmierung unter Basic, über höhere Programmiersprachen, Problemlösungen, Textverarbeitungs-Programmen bis hin zu Spielen, macht der Joyce überall mit, solange es keinen mehrfarbigen Bildschirm erfordert. Die mitgelieferte Software ist sehr reichhaltig und leistungsfähig, die Hardware nahezu vollständig.

Der Joyce steht also für ein komplettes System, bei dem man in Anbetracht des Preises nur staunen kann.

Jedem, der nicht unbedingt den neuesten Stand der Technik, sprich 16/32 Bit, sein eigen nennen muß, wird der Joyce eine preiswerte Hilfe zur Erledigung seiner täglichen Arbeiten sein. Denn preiswert heißt noch lange nicht billig! (Jörg Höhnerbach/ja)

Joyce: mehr als e

Lernen Sie, mehr aus Ihrem Joyce herauszuholen als Texte. Setzen Sie ihn als vollwertigen Personal Computer ein.

Da die Werbung vom Joyce immer als Textverarbeitungssystem spricht, liegt die Annahme nicht fern, er sei nur eine bessere Schreibmaschine mit Bildschirm. Diejenigen, die sich mit dem Betriebssystem CP/M und dem Basic-Interpreter MBasic auskennen, haben sicherlich schnell die über Textverarbeitung hinausgehenden Fähigkeiten erkannt. Allen, die sich in der Vergangenheit noch nicht mit dem Programmieren in Basic auseinandergesetzt haben, sollen hier eine Einführung in das mitgelieferte Mallard-Basic 80 erfahren.

Viele Probleme, ob einfacher oder komplexer Art, lassen sich im täglichen Gebrauch wesentlich schneller bewältigen, wenn man sie mit einem Computer bearbeitet. Dieses Programm kann man speichern und je nach Bedarf aufrufen. Damit der Joyce in der Lage ist, Basic zu verstehen, muß zuerst der Interpreter geladen werden. Dazu schalten Sie Ihren Computer an und legen die Seite 2 der mitgelieferten Disketten ein. Möchten Sie Basic nach dem Arbeiten mit Loco-Script laden, verwenden Sie ebenfalls die Seite 2 der Systemdiskette und drücken gleichzeitig die Tasten SHIFT, EXTRA und EXIT. In beiden Fällen erscheint nach kurzer Zeit eine Meldung über die CP/M Plus Version und das Bereitschaftszeichen »A>«, auch Prompt genannt, auf dem Bildschirm. Der Befehl »DIR« listet nun das Inhaltsverzeichnis der Diskette auf dem Bildschirm auf. Suchen Sie die Datei mit dem Namen BASIC.COM. Tippen Sie nun »BASIC« ein, denn dies teilt dem System mit, daß Sie den Basic-Interpreter laden wollen. Nach einigen Sekunden meldet sich dann Mallard-Basic 80 mit einer Bemerkung über den freien Speicherplatz (»31597 free bytes«). Nun steht der Arbeit mit Basic nichts mehr im Wege.

Unter Basic kann man grundsätzlich alle Befehle direkt einzugeben. Der Computer bearbeitet sie dann nach Betätigen der RETURN-Taste sofort. Da der Vorteil eines Computers jedoch in der Zusammenfassung mehrerer Befehle zu einem Programm besteht, beginnen wir mit einem Progammbeispiel. Sie können jedoch auch einmal probieren, die Befehle einzeln (ohne Zeilennummer) einzugeben. Beobachten Sie das Ergebnis. Tippen Sie nun das Programm aus Listing 1 ein; zuerst die Zeilennummer, dann den Befehl. Jeden Befehl (beispielsweise PRINT) können Sie auch mit kleinen Buchstaben eingeben, der Computer wandelt diesen dann selbständig in Großbuchstaben um. Die Reihenfolge der Programmzeilen spielt dabei keine Rolle. Geben Sie die Zeilen einfach durcheinander ein. Mit dem Befehl »LIST« wird das bisher eingegebene Programm auf dem Bildschirm anzeigt. Und zwar sortiert Joyce sie numerisch aufsteigend.

Unser Programm berechnet den Preis eines Artikels (Verkaufspreis 150 Mark) ohne Mehrwertsteuer und die Mehrwertsteuer (14 Prozent), die im VK (Verkaufspreis) enthalten ist.

Das Komando PRINT in Zeile 10 bewirkt die Ausgabe des Textes zwischen den Anführungszeichen auf dem Bildschirm. Anders in Zeile 20: Hier wird zuerst der Wert des Terms hinter der PRINT-Anweisung berechnet und dann ausgegeben. Gleiches geschieht in Zeile 30. Beim Aufstellen von Berechnungstermen gilt die allgemein bekannte mathematische Hierarchie: Punktrechnung vor Strichrechnung. Also eventuelle Klammern nicht vergessen! Zeile 40 teilt dem Computer mit, daß das Programm an dieser Stelle endet. Bei solch einfachen Programmen ist diese Zeile nicht unbedingt erforderlich, bei komplexeren Programmen jedoch unbedingt notwendig, wie Sie später sehen.

Sie starten das Programm mit dem Befehl »RUN«, gefolgt von der Taste RETURN. Der Befehl RUN veranlaßt das Verlassen des direkten Eingabemodus, um das im Speicher befindliche Programm zu bearbeiten. Erscheint die Ausgabe »SYNTAX ERROR« auf dem Bildschirm, haben Sie einen Befehl falsch geschrieben. Bei den später verwendeten Programmen ist es auch möglich, daß Sie anstelle eines Kommas einen Doppelpunkt gesetzt haben. Um die entsprechende Zeile zu ändern,

```
10 PRINT"Mehrwertsteuer und Nettopreis a
us dem Verkaufspreis"
20 PRINT 150/1.14
30 PRINT 150*0.14/1.14
40 END
```

Listing 1. Aus einem Grundgerüst...



Schreibmaschine!

geben sie den Befehl »EDIT Zeilennummer«, gefolgt von der RETURN-Taste, ein. Die zu korrigierende Zeile erscheint nun auf dem Monitor, und mit der DEL-Taste lassen sich die betreffenden Zeichen löschen, nachdem Sie den Cursor (Positionsanzeiger) mit den Pfeiltasten an die entsprechende Stelle gebracht haben. Nun können Sie die richtigen Zeichen einsetzen.

Während der Programm-Bearbeitung werden sowohl der Text aus Zeile 10 wie auch die berechneten Werte auf dem Bildschirm ausgegeben. Die Bemerkung »OK« besagt, daß das gesamte Programm vom Computer bearbeitet und für neue Eingaben oder eventuelle Änderungen bereit ist. Das erzielte Ergebnis ist aber in vielerlei Hinsicht unbefriedigend. So können Sie beispielsweise nicht verschiedene Verkaufspreise bearbeiten, ohne direkt in das Programm eingreifen zu müssen. Wir wollen nun nach und nach Verbesserungen vornehmen.

Geben Sie bitte folgende zusätzliche Zeile ein:

```
15 input vk
```

Der INPUT-Befehl stoppt das Programm bei Zeile 15 und wartet auf eine Eingabe. Dies zeigt der Computer durch ein Fragezeichen am Bildschirm an. Der eingegebene Wert wird dann in der Variablen vk (Abkürzung für Verkaufspreis) gespeichert. Variablen sind Speicherplätze, auf die Sie später nach Bedarf zurückgreifen können. Sowohl Zahlen als auch Text sind zulässige Eingabewerte für Variablen. Sie muß aber zur Aufnahme des entsprechenden Ausdrucks vorbereitet sein. Soll eine Ganzzahl (1, 2, 5, 456, 1025 und so weiter) gespeichert werden, ohne Speicherplatz zu vergeuden, muß hinter dem Variablennamen ein »%« stehen: zum Beispiel »vk%«. Bei diesen Integervariablen wird weniger RAM-Bereich gebraucht. Dadurch verkürzt sich auch die Rechenzeit, was bei längeren Programmen einen deutlichen Zeitgewinn zur Folge haben kann. Versucht man jedoch, eine Gleitkommazahl zu speichern (0,1, 0,022, etc.), gehen die Nachkommastellen verloren. Ein Variablenname ohne Zusatz steht immer für eine Gleitkommazahl.

Der Zusatz »$« ermöglicht das Speichern von Zeichenketten bis zu 255 Zeichen (zum Beispiel »vk$«). So gespeicherte Zahlen stehen nicht ohne weiteres für Berechnungen zur Verfügung, da sie das Basic als Zeichenfolge betrachtet. Eine Spezialität des Mallard Basic 80 (und MBasic) ist die Fähigkeit, Zahlen mit doppelter Genauigkeit zu verarbeiten. Normalerweise werden Berechnungen nur auf sieben Stellen hinter dem Komma genau durchgeführt. Für spezielle Anwendungen steht durch die doppelte Genauigkeit ein Mittel zur Verfügung, auf 16 Stellen hinter dem Komma genau zu arbeiten. Dazu muß ein »#« hinter dem Variablennamen stehen. (Achtung! doppelt genaue Berechnung erfordert doppelt genaue Ausgangswerte! Das belegt natürlich mehr Speicherplatz und verlängert die Rechenzeit.).

Fassen wir zusammen; es gibt grundsätzlich 4 Arten von Variablen:

1. Ganzzahl Name% (Werte von −32768 bis +32767)
2. Gleitkomma Name! (Werte von absolut 1.7E+38 bis 2.9E-37)
3. Doppelte Genauigkeit Name # (wie 2. jedoch doppelt genau)
4. Zeichenketten Name$

Die Bezeichnung der Variabeln läßt durchaus Anschaulichkeit zu, wie zum Beispiel »verkaufspreis« oder »einnahme«. Jedoch sollten Sie sich, um Schreibarbeit zu sparen, so kurz wie möglich (aber dennoch verständlich) fassen. Beachten Sie: Das erste Zeichen einer Variable muß ein Buchstabe sein!

Zurück zu unserem Programm. In Zeile 20 und 30 ersetzen wir jetzt die Zahl 150 durch vk, da diese Variable den Verkaufspreis beinhaltet. Durch den Befehl »EDIT 20« erhalten Sie die Zeile 20, und die notwendige Änderung läßt sich schnell erledigen (erst löschen, dann ersetzen, schließlich mit RETURN bestätigen). Vergleichen Sie Ihre Eingaben mit Listing 2.

Nun starten Sie das Programm mit »RUN«. Erscheint das Fragezeichen, geben Sie den Verkaufspreis ein. Wie Sie sehen, ist diese Programm-Version schon universeller (nämlich für unterschiedliche Verkaufspreise) einsetzbar. Doch stellen Sie sich vor, es sollte jemand mit dem Programm arbeiten, der es nicht kennt. Sicherlich wird ihn das Fragezeichen vor größere Verständnisprobleme stellen. Ebenso stört es, das Programm jedesmal neu starten zu müssen. Ändern Sie also das Programm wie in Listing 3.

Wie Sie sehen ist es möglich, Texteingaben und Variablen zu einer PRINT-Anweisung zusammenzufassen.

Ein Semikolon trennt diese Textpassagen von den Variablen. Vergessen Sie dieses Zeichen, erfolgt ein »SYNTAX ERROR«. Versuchen Sie einmal, bei der INPUT-Anweisung, das Semikolon gegen ein Komma auszutauschen und beachten Sie den Unterschied im Programmablauf: Das Fragezeichen wird unterdrückt.

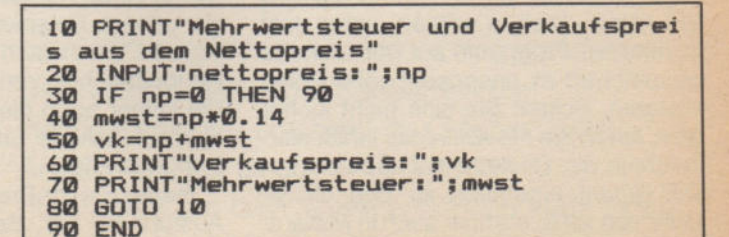
Wenn Sie diese Version starten, erhält die INPUT-Anweisung eine Erklärung des erwarteten Eingabewertes, die Ausgaben werden bezeichnet und Zeile 80 veranlaßt einen Neubeginn des Programms. Der Befehl »GOTO (Zeilennummer)« läßt das Programm zu der Zeile verzweigen, die die Zeilennummer festlegt. Die GOTO-Anweisung beeinflußt den gradlinigen Ablauf eines Programms.

Wir benötigen nun die END-Anweisung, da das Programm sonst zu einer nicht vorhandenen Zeilennummer verzweigt. Und das hat eine Fehlermeldung zur Folge. Wenn Sie nun für den Nettopreis eine 0 eingeben, unterbricht das Programm, da es zu Zeile 90 (hier steht die END-Anweisung) verzweigt. Mit der IF.....THEN-Anweisung lassen sich also Bedingungen abfragen (Ist der Wert für np = 0, dann mach das, was hinter der THEN-Anweisung steht). Ist die IF-Bedingung nicht erfüllt, setzt das Programm in der nächsten Zeile den Ablauf fort. Die IF...THEN-Anweisung läßt sich noch durch IF...THEN...ELSE erweitern. Das Programm führt bei Erfüllung der Bedingung den Teil hinter der THEN-Anweisung durch, ansonsten die Anweisung hinter ELSE. Experimentieren Sie ein wenig mit dem Programm.

Mittlerweile haben Sie sicher einige Befehle zwischen die üblichen Zehnerschritte eingefügt. Der alte Abstand läßt sich jedoch ohne Schwierigkeiten wie-

```
10 PRINT"Mehrwertsteuer und Nettopreis a
us dem Verkaufspreis"
15 INPUT vk
20 PRINT vk/1.14
30 PRINT vk*0.14/1.14
40 END
```

Listing 2. ...wird nach und nach...



```
10 PRINT"Mehrwertsteuer und Verkaufsprei
s aus dem Nettopreis"
20 INPUT"nettopreis:";np
30 IF np=0 THEN 90
40 mwst=np*0.14
50 vk=np+mwst
60 PRINT"Verkaufspreis:";vk
70 PRINT"Mehrwertsteuer:";mwst
80 GOTO 10
90 END
```

Listing 3. ...durch neue Befehle...

```
10 PRINT"Berechnung der Mehrwertsteuer a
us vk oder Netto fuer mehrere Artikel"
20 INPUT"Wieviele Artikel:";anzahl
30 mwst.gesamt=0:vk.gesamt=0:np.gesamt=0
40 FOR i=1 TO anzahl
50 INPUT"Grundlage VK...->1 oder NP...->
2";auswahl
60 IF auswahl=1 THEN GOSUB 1000
70 IF auswahl=2 THEN GOSUB 2000
80 IF auswahl<>1 AND auswahl<>2 THEN 50
90 mwst.gesamt=mwst.gesamt+mwst
100 np.gesamt=np.gesamt+np
110 vk.gesamt=vk.gesamt+vk
120 NEXT i
130 PRINT:PRINT:PRINT
140 PRINT"Die Artikel haben einen Gesamt
nettopreis von:";np.gesamt;" DM"
150 PRINT"Die anfallende Mehrwertsteuer
betraegt:";mwst.gesamt;" DM"
160 PRINT"Daraus ergibt sich ein Gesamtv
erkaufspreis von:";vk.gesamt;" DM"
170 END
1000 INPUT"Verkaufspreis:";vk
1010 mwst=vk*0.14/1.14
1020 np=vk-mwst
1025 PRINT vk,mwst
1030 RETURN
2000 INPUT"Nettopreis:";np
2010 mwst=np*0.14
2020 vk=np+mwst
2030 RETURN
```

Listing 4. ...ein immer besseres...

```
10 PRINT"Berechnung der Mehrwertsteuer a
us vk oder Netto fuer mehrere Artikel"
20 INPUT"Wieviele Artikel:";anzahl
30 mwst.gesamt=0:vk.gesamt=0:np.gesamt=0
40 FOR i=1 TO anzahl
50 INPUT"Grundlage VK...->1 oder NP...->
2";auswahl
60 IF auswahl <> 1 AND auswahl <>2 THEN
50
70 ON auswahl GOSUB 1000,2000
80 mwst.gesamt=mwst.gesamt+mwst
90 np.gesamt=np.gesamt+np
100 vk.gesamt=vk.gesamt+vk
110 NEXT i
120 PRINT:PRINT:PRINT
130 PRINT"Die Artikel haben einen Gesamt
nettopreis von:";np.gesamt;" DM"
140 PRINT"Die anfallende Mehrwertsteuer
betraegt:";mwst.gesamt;" DM"
150 PRINT "Daraus ergibt sich ein Gesamt
verkaufspreis von:";vk.gesamt;" DM"
160 END
1000 INPUT"Verkaufspreis:";vk
1010 mwst=vk*0.14/1.14
1020 np=vk-mwst
1030 RETURN
2000 INPUT"Nettopreis:";np
2010 mwst=np*0.14
2020 vk=np+mwst
2030 RETURN
```

Listing 5. ...Programm zur Mehrwertsteuerberechnung

derherstellen. Versuchen Sie aber nicht, jede einzelne Zeilennummer zu ändern, denn dies erledigt der Befehl RENUM. Dieser Befehl veranlaßt eine neue Zeilennumerierung, beginnend mit 10. Dabei werden alle GOTO-Befehle natürlich auch entsprechend geändert. Geben Sie »RENUM« ein und lassen dann das Programm neu auflisten. Wünschen Sie eine bestimmte Anfangszeile, so geben Sie diese hinter dem Befehl ein (beispielsweise »RENUM 100«). Ist ein anderer Abstand gewünscht, muß er durch ein Komma getrennt hinter der Startzeilennummer stehen. Der Befehl »RENUM 200,40« hat zur Folge, daß Ihr Programm anschließend mit Zeile 200 beginnt und einen Zeilenabstand von 40 hat.

Wenn Sie mit der Routine später noch einmal arbeiten möchten, sollten Sie es auf einer Diskette speichern. Der entsprechende Befehl lautet: »SAVE-"name"«. Der Name zwischen den Anführungszeichen darf maximal acht Zeichen vor, dann einen Punkt und drei Zeichen nach dem Punkt enthalten. Lassen Sie die Stellen hinter dem Punkt frei, fügt der Computer automatisch »BAS« für Basic an. Beachten Sie, daß Sie keinen Namen wählen, unter dem bereits ein Programm auf der Diskette gespeichert ist, ansonsten geht dieses verloren. Sollten Sie sich nicht sicher sein, listen Sie mit »DIR« das Inhaltsverzeichnis der Diskette auf. Obwohl der DIR-Befehl eigentlich zu den CP/M-Befehlen zählt, steht er auch in Mallard-Basic zur Verfügung.

Wollen Sie das Programm wieder in den Speicher des Joyce laden, geben Sie den Befehl »LOAD"name"« ein. Die Stellen nach dem Punkt können Sie vernachlässigen.

Bisher haben wir die Ergebnisse immer unmittelbar vor der Ausgabe durch die PRINT-Anweisung berechnet. Dies bedeutet, daß die so ermittelten Werte nur für diese eine Ausgabe zur Verfügung stehen.

Sicherlich sind die hier aufgeführten Problemstellungen recht simpel, doch geht es ja in erster Linie um das grundsätzliche Verständnis der Progammierung. Machen Sie sich immer die erforderlichen Rechenschritte klar, wobei Sie sich bemühen sollten, einen Weg zu finden, die Ergebnisse vor der Ausgabe einem Speicherplatz (Variable) zuzuweisen. Dies hat den Vorteil, ohne erneute Berechnung der Werte diese trotzdem zur Verfügung zu haben.

Bevor Sie die neue Routine eingeben, müssen Sie den Programmspeicher des Joyce löschen. Die Anweisung SAVE hat nur den aktuellen Programmstand gespeichert, ohne das Programm aus dem Speicher zu entfernen. Dies ermöglicht Ihnen ein Zwischenspeichern Ihrer Programme, die häufig einen weitaus größeren Umfang als unsere Beispielprogramme haben werden. Wenn dann einmal die Daten im Hauptspeicher verlorengehen, haben Sie immer noch die gespeicherte Version auf Diskette, und der Schaden hält sich in Grenzen.

Testen Sie Ihr Programm auf richtige Ausführung hin. Bedenken Sie, nicht nur Sie sollen das Programm richtig handhaben können, sondern auch Fremdbenutzer. Ein gewisses Maß an Komfort ist grundsätzlich nicht zu verachten.

Nehmen wir nun an, Sie sollen für verschiedene Artikel, von denen Sie einmal den Netto-, einmal den Verkaufspreis wissen, jeweils Nettopreis, Verkaufspreis und Mehrwertsteuer berechnen. Bei dieser Problemstellung kann sowohl das erste wie auch das zweite Programm zur Anwendung kommen. Sie laden dazu also immer das jeweils erforderliche Programm in den Speicher, um die entsprechende Berechnung durchzuführen. Dies ständige Laden ist sicherlich nicht komfortabel. Deshalb scheint es günstig, die Programme miteinander zu verbinden. Wie schon erwähnt, gibt es außer GOTO weitere Anweisungen, in Programme eine Verzweigung einzubauen. Eine davon ist die GOSUB-Anweisung. Sie bewirkt eigentlich das gleiche wie »GOTO«. Nur erhält die Verzweigungszeile bei Verwendung von GOSUB eine Marke, zu der das Programm zurückkehrt, sobald es auf den Befehl RETURN stößt. Das heißt, das Programm fährt in der Zeile nach der GOSUB-Anweisung fort, sobald in der Befehlsfolge nach der Verzweigungsdresse ein RETURN erreicht wird. Diese Programmteile erhalten meist Zeilennummern, die mehr oder weniger weit unterhalb (größere Zeilennummern) des eigentlichen Programms liegen (nicht zwingend erforderlich, aber zweckmäßig). Deshalb spricht man auch von Unterprogrammen. Diese Art der Programmierung vermeidet, bestimmte Befehlsfolgen doppelt oder

mehrfach in das Programm aufzunehmen. Bei Bedarf verzweigt einfach eine GOSUB-Anweisung zu dem betreffenden Unterprogramm.

Gehen wir weiter zu Listing 4. Die Kopfzeile gibt wieder den Verwendungszweck des Programms auf dem Bildschirm aus. Darauf folgt die Frage nach der gewünschten Berechnung (Taste »1« oder »2« gedrückt?). Die IF-Abfragen stellen sicher, daß zu dem richtigen Unterprogramm verzweigt wird. Die dritte IF-Abfrage verhindert eine Fehleingabe durch eine Rückkehr zum Auswahlmenü. Beachten Sie den logischen Operator AND, der eine weitere IF-Abfrage überflüssig macht. Bedeutung: Wenn Auswahl ungleich 1 und Auswahl ungleich 2, dann... Die Unterprogramme können hinter die END-Anweisung gesetzt werden. Nach dem Unterprogramm folgt die Ausgabe der berechneten Werte (im Unterprogramm erfolgt nur eine Zuweisung zu einer Variablen). Die REM- Anweisung hat auf das Programm keinerlei Auswirkungen. Denn alles, was hinter einer REM-Anweisung steht, dient nur dazu, ein Programm übersichtlicher zu gestalten und zu kommentieren. Dasselbe gilt für die Zeilen, die nur einen Doppelpunkt enthalten. Mit diesen Zeilen schaffen Sie einen Abstand zu den nachfolgenden Zeilen, und das Programm läßt sich nach einer LIST-Anweisung leichter lesen.

Probieren Sie das Programm einmal aus. Vergessen Sie aber nicht, den Speicher Ihres Joyce vor der Eingabe mit »NEW« zu löschen.

Durch eine Erweiterung des GOSUB-Befehls läßt sich das Programm verkürzen. Löschen Sie dazu die beiden ersten IF-Abfragen und fügen Sie hinter der dritten IF-Abfrage eine Zeile mit dem Befehl »ON auswahl GOSUB 1000,2000« ein. Er hat zur Folge, daß bei der Eingabe einer Eins zur Zeile 1000 verzweigt wird, bei der Eingabe einer Zwei analog zur Zeile 2000. Die Liste der Unterprogrammstartzeilen nach dem GOSUB kann man noch erheblich erweitern. Dasselbe gilt auch für den Befehl GOTO, aber bedenken Sie, dort erfolgt keine automatische Rückkehr.

Eine weitere Problemstellung ist, die Verkaufspreise, Nettopreise und Mehrwertsteuer mehrerer Artikel zu berechnen und aufzusummieren, um beispielsweise einen Kostenvoranschlag zu erstellen. Hierzu führen Sie die Berechnung für eine bestimmte Anzahl von Artikeln durch. Das Aufschreiben und Zusammenrechnen der Einzelergebnisse von Hand ist aufwendig. Basic stellt auch für dieses Problem einen Befehl (Mallard-Basic sogar zwei) zur Verfügung; eine

1. FOR-TO-NEXT-Schleife oder
2. WHILE-WEND-Schleife

Der erste Befehl empfiehlt sich, wenn die Anzahl der zu berechnenden Artikel bereits zu Beginn der Programmausführung feststeht. Betrachten Sie Listing 5. Die Anzahl der Artikel wird durch eine INPUT-Anweisung abgefragt und anschließend einem Speicher zugewiesen. Die Speicher für die Summenbildung werden dann auf Null gesetzt, um zu verhindern, daß bei einem erneuten Durchlauf nach einer GOTO-Anweisung die Speicher noch durch alte Werte belegt sind. Speicher werden ansonsten nur nach folgenden Anweisungen auf Null gesetzt: RUN und LOAD. Im Programm beginnt nun die Schleife mit der Anweisung »FOR I=1 to anzahl«. Die Anweisung »NEXT I« schließt die Schleife ab. Die Befehle zwischen diesen Zeilen wiederholen sich nun so oft, bis der Speicher I den Wert von »anzahl« überschreitet. Dazu erhöht sich nach jedem Durchlauf I um den Wert 1. Anschließend setzt das Programm den Ablauf in der nächsten Zeile fort. Geben Sie auch dieses Programm ein und verfolgen Sie den Ablauf (vergessen Sie den Befehl NEW vor der Eingabe des Programms nicht).

Der Schleifenzähler, hier die Variable I, ist auch direkt als Rechenvariable zu nutzen.

Ist die Anzahl der Artikel zu Beginn des Programmablaufs nicht bekannt, verwendet man den zweiten Befehl. Ersetzen Sie »FOR..TO« durch »WHILE a$<>"j"« und die NEXT-Anweisung durch »WEND«. Fügen Sie unmittelbar vor »WEND« den Befehl »INPUT "Abbruch ?(j)";a$« ein. Die Schleife wird nun so oft durchlaufen, bis die WHILE-Anweisung falsch ist. Solange das Zeichen in a$ ungleich »j« ist, wird die Schleife erneut bearbeitet. Möchten Sie diesen Vorgang abbrechen, geben Sie auf die Frage "Abbruch ?(j)" »j« ein. Auf jede andere Eingabe erfolgt ein weiterer Durchlauf. Das »j« in der WHILE-Anweisung muß auf jeden Fall in Anführungszeichen stehen, da es sich ja um ein Zeichen handelt!

Mit den bis hierher besprochenen Befehlen können Sie bereits eigene Programme schreiben. Stellen Sie sich selbst einige Aufgaben, zum Beispiel: Wieviel muß ein Artikel kosten, wenn er nach Abzug der Vorsteuer eine Gewinnspanne von 35 Prozent haben soll? Oder, berechnen Sie die Fakultät einer Zahl (5!=1*2*3*4*5).

Wächst der Umfang Ihrer Programme, sparen Sie sich das Eingeben der Zeilennummern, wenn Sie zuerst den Befehl »AUTO« eingeben. Es erscheint tatsächlich die Zeilennummer 10 und der Cursor zeigt an, daß Sie nun mit der Eingabe beginnen können. Beim Abschicken der Zeile erscheint die nächste Zeilennummer im 10er-Abstand. Wünschen Sie eine andere Startzeile und einen anderen Zeilenabstand, schreiben Sie dies hinter den Befehl. »AUTO 500,20« bewirkt den Beginn der Zählung bei Zeile 500 in 20er-Schritten.

Ein weiterer Befehl, der das Programmieren vereinfacht, heißt »DELETE«. Dieser Befehl entfernt bestimmte Zeilen (Bereiche) aus Ihrem Programm »DELETE 20-200«, löscht die Zeilen 20 bis einschließlich 200.

Schließlich gibt es noch die Möglichkeit, alles was auf dem Bildschirm erscheint, auf dem Drucker zu dokumentieren. Ersetzen Sie dazu nur die PRINT-Befehle durch »LPRINT«. Das Erstellen von Listings erfolgt auf den Befehl »LLIST«.

Die hier aufgeführten Befehle stellen nur einen kleinen Teil des Mallard-Basic-Befehlsvorrats dar und entsprechen bis auf einige Ausnahmen (WHILE,WEND,ELSE) dem Standard-Basic. Die wohl wesentlichste Neuerung ist die Dateiverwaltung mit »JETSAM«, welche eine Bearbeitung mit Schlüsselwerten zuläßt.

Ist Ihr Interesse geweckt und möchten Sie beginnen, sich eine Programmbibliothek anzulegen, arbeiten Sie im Dialog mit dem Computer das mitgelieferte Basic-Handbuch durch. Lassen Sie sich dabei nie entmutigen, auch wenn dies manchmal zu einer abenteuerlichen Schatzsuche ausartet.

(Jörg Höhnerbach/ja)



Joyce-Requisiten

Für den Joyce war schon kurz nach seiner Einführung ein erfreulich reichhaltiges Softwareangebot verfügbar. Unsere große Übersicht zeigt Ihnen, welche Programme es gibt.

Ein Computer ist soviel wert, wie sein Softwareangebot. Nach dieser Formel schneidet der Joyce sehr gut ab. Bei professionellen Anwendungen, aber auch bei der Unterhaltungssoftware, decken die Anbieter ein breites Spektrum ab. Unsere Übersicht erhebt keinen Anspruch auf Vollständigkeit. Alle Daten beruhen auf Angaben der Anbieter.

(Matthias Rosin)

Programmname	Anwendung	Preis DM	Anbieter	Besonderheiten
Business-Software				
Adresscomp	Adreßverwaltung	58,-	ZA	-
Auftrag-Fakt	Auftrags-Rechnungswesen	415,-	KH	-
Business-Pack	Integriertes Paket	199,-	ZS	Fakturierung, Adreß-Manager, Lager-Manager
Caltex	Textverarbeitung	2736,-	VS	mit Datenverwaltung und Kalkulation
ComPack	Integriertes Paket	798,-	SR	Lagerbestandsführung, Auftragsbearbeitung, Fakturierung, Finanzbuchhaltung
Cursy	Integriertes Paket	998.-	VS	Datenbank, Textverarbeitung, Serienbriefe
Data Base	Toolbox	225.-	HS	Turbo-Pascal erforderlich; zum Erstellen von Dateiverwaltungsprogrammen; Quelltext vorh.
Datei-Star	Dateiverwaltung	98,-	ST	mit integrierter Kalkulation
dBase II	Datenbanksystem	199,-	MT	-
DR Draw	Zeichenprogramm	199,-	MT	für technische Zeichnungen, Flußdiagramme, Karten, Schaubilder etc.
DR Graph	Präsentationsgrafik	199,-	MT	für grafische Auswertungen jeder Art; mit Softwareschnittstelle zu Multiplan
Easyfakt	Fakturierung	500,-	GE	-
Easyfibu	Finanzbuchhaltung	500,-	GE	-
Fabrik	Materialwirtschaft	345,-	KH	-
Fakt	Fakturierung	285,-	KH	-
Fakten	Geschäftsprogramm	198,-	WE	für Rechnungen, Lieferscheine, Etiketten, Mahnungen, Kundendatei, Textverarbeitung etc.
FaktuRAM	Fakturierung	98,-	ZA	-
Fakturierung	Fakturierung	94,-	MT	dBase II-Anwenderprogramm mit dokumentiertem Quellcode
Fakturierung-Auftragsbearbeitung	Fakturierung	171,-	SC	-
Fibu	Finanzbuchhaltung	622,-	KH	-
Fibu-King	Finanzbuchhaltung	136,-	ZA	-
Fibu Star Plus	Finanzbuchhaltung	298,-	ST	in Cobol geschrieben
Finanz-Buchhaltung	Finanzbuchhaltung	94,-	MT	Kontenpläne, Einnahmen-/Überschußrechnung, Journalschreibung etc.
Finanzbuchhaltung	Finanzbuchhaltung	1140,-	SC	-
Finanzwunder-Mercator	Buchhaltung	477,-	CO	-
Hausverwaltung	Büroanwendung	570,-	SC	-
Hercules I	Datenbank	298,-	KH	-
Joyce-Mailing-System	Programmpaket	189,-	ST	entspricht Datei-Star und Star-Mail
Kalkulation	Kalkulationsprogramm	345,-	KH	-
Kassenbuch-Tagesabrechung	Büroanwendung	171,-	SC	-
Lagdat	Lagerverwaltung	68,-	ZA	-
Lager	Lagerhaltung	345,-	KH	-
Lohnbuchhaltung	Lohnbuchhaltung	1140,-	SC	-
Lohnwunder-Pecunia	Lohnabrechnung	577,-	CO	-
Multiplan 1.06	Tabellenkalkulation	199,-	MT	
ProfiRAM	Integriertes Paket	178,-	ZA	entspricht Adresscomp, Lagdat und FaktuRAM
Prompt	Datei-/Maskengenerator	69,-	SR	frei programmierbare Dateiverwaltung mit Listengenerator und Sortierprogramm
Prompt-Druck	Druckerprogramm	49,-	SR	verknüpft die Textverarbeitung LocoScript mit der Dateiverwaltung Prompt; druckt Serienbriefe, Adreßaufkleber etc.
Quickdata +	Datenverwaltung	198,-	KH	-
Quickfakt +	Fakturierung	198,-	KH	-
Quickfibu +	Finanzbuchhaltung	198,-	KH	-
Quickmask +	Formulargenerator	198,-	KH	-
Quickwrite +	Textverarbeitung	198,-	KH	-
RH-Data	Dateiverwaltung	89,-	IH	max. 1003 Datensätze
Sampler I	Integriertes Paket	79,90	VA	Autokostenrechnung, Diätplan, Videoverwaltung, Zins/Tilgungsplan, Fußballtabelle, Biorhythmus
Stamm	Stammdatenverwaltung	285,-	KH	-
Star-Mail	Druckerprogramm	98.-		verknüpft die Textverarbeitung LocoScript mit den Daten aus Datei-Star; Serienbriefausgabe; Fremddruckerausgabe
SuperCalc2	Kalkulationsprogramm	198,-	SR	englische Anleitung
Text	Textverarbeitung	298,-	KH	-
Turbo Adress	Text-/Adreßverwaltung	169,-	IH	für Rundschreiben, Etiketten, Listendruck; max. 1400 Datensätze
Turbo-Lader Business	Programmbibliothek	148,-	MT	Routinen zur Dateiverwaltung im TURBO-Pascal-Quellcode
VSC-Fibu	Finanzbuchhaltung	2736,-	VS	-
VSC-Text	Textverarbeitung	364,80	VS	-
WordStar 3.0	Textverarbeitung	199,-	MT	mit MailMerge
Programmiersprachen				
C-Basic-Compiler	Allgemeine Anwendungen	174,-	MT	-
Hisoft C-Compiler	Allgemeine Anwendungen	138,90	PS	-
Nevada Cobol	Kaufm. Anwendungen	189,-	SD	-
Nevada Fortran	Naturwissenschaftl. Anw.	189,-	SD	-
Pascal/MT+	Allgemeine Anwendungen	174,-	MT	-
Small-C	Allgemeine Anwendungen	148,-	MT	enthält Compiler, Linker, Assembler, Editor, Tools
TURBO-Pascal	Allgemeine Anwendungen	225,72	MT	sehr schneller Pascal-Compiler
TURBO-Toolbox	Programmbibliothek	225,72	MT	-
TURBO-Tutor	Lehrprogramm	104,86	MT	benötigt TURBO-Pascal
Wissenschaft und Technik				
Finanzmathematik	Mathematik	95,-	FI	Zinseszinsrechnung, Wechseldiskontierung, Abschreibung, Rentenrechnung, Kursrechnung, Rentabilität
MICA	CAD-Programm	198,-	EC/IH	maßgerechte Konstruktion in sechs Ebenen, Symbol-Bibliotheken, Plotter-/Druckerausgabe
PAD Plus	Statistik	148,-	ES	Regression, Balkendiagramm, Druckerausgabe
Statistik	Statistik	345,-	KH	-
TURBO-Lader-Grundpaket	Programmbibliothek	138,-	MT	mathematische Funktionen, Diskettenroutinen, Bitmanipulation, Sortierverfahren etc.
TURBO-Lader-Science	Programmbibliothek	189,-	MT	Statistik für Technik, Betriebswirtschaft, Medizin; Quellcode in TURBO-Pascal vorhanden
Spiele				
Bridge	Kartenspiel	79,90	PS	-
Heroes of carn	Grafikadventure	90,-	PS/RU	ist identisch mit der Version vom C 64
3D-Clock-Chess	Schach	69,-	ZS/PS/RU	spielstark
Ping Pong	Geschicklichkeitsspiel	149,-	ZS	-
Sonstiges				
Vokabel-Trainer	Lernprogramm	59,-	ES	C-Basic compiliert

CO Computer-Studio-Offenbach
Kl. Biergrund 17, 6050 Offenbach, 069/810613

EC E&C Rupert Zellmeier
Dompfaffstr. 127a, 8520 Erlangen, 09131/440303

ES Escon
Am Rindermarkt 4a, 8050 Freising, 08161/13089

FI Dr. Jürgen Fiedler
Am Tönnessenkreuz 5, 5300 Bonn 1, 0228/649240

GE Gruse Elektronik
Stieglitzweg 1, 3300 Braunschweig, 0531/352300

HS Heimsoeth-Software
Fraunhoferstr.13, 8000 München 5, 089/264060

IH Integral Hydraulik
Am Hochofen 108, 4000 Düsseldorf 11, 0211/5065-213

KH KHS-Software
Heidemannstr. 1, 8000 München 45, -

MT Markt&Technik
Hans-Pinsel-Str. 2, 8013 Haar, 089/4613-0

PS Profisoft
Sutthauserstr. 50/52, 4500 Osnabrück, 0541/55488

RU Rushware
An d. Gümpgesbrücke 24, 4044 Kaarst 2, 02101/68499

SC Walter Schloßmacher
Tannenstr. 12, 4048 Grevenbroich, 02181/63095

SD Schneider Data
Rindermarkt 8, 8050 Freising, 08161/2877

ST Star Division
Zum Elfenbruch 1, 2120 Lüneburg, 04131/46093

VA Valc Peter Kohl
Waggasse 4, 8230 Bad Reichenhall, 08651/66773

VS VSC-Datasysteme
Austr. 34, 7730 VS-Schwenningen, 07720/61029

WE Weeske Computer-Elektronik
Potsdamer Ring, 7150 Backnang, 07191/1528-29

ZA Van-der-Zalm-Software
Schieferstätte, 2949 Wangerland 3, 04461/71719

ZS ZS-Soft
Postf. 2361, 8240 Berchtesgaden, 08652/63061-2691

Basic: übersichtlich wie Pascal

Auch Ihr Basic-Interpreter kennt Marken und die REPEAT-UNTIL-Anweisung. Sie brauchen nur ein passendes Hilfsprogramm.

Kennzeichnend für viele Basic-Programme ist Unübersichtlichkeit. Zeilennummern und Befehle wie GOTO und GOSUB verleiten jeden Programmierer dazu, mehr oder weniger sinnvolle Sprünge in eine Routine einzufügen, wenn der Platz eng wird. Strukturierung ist für die meisten Basic-Fans ein Fremdwort.

Gegenüber den ersten Basic-Interpretern kennt das Mallard- und das Locomotiv-Basic der Schneider-Computer allerdings schon viele neue strukturierte Anweisungen. Besonders die WHILE-WEND-Schleife erlaubt, Programmteile ohne Sprungbefehle zu verwenden. Es fehlt aber immer noch der von Pascal her beliebte Befehl REPEAT-UNTIL. Diesen und Sprunganweisungen auf Marken – und nicht an eine bestimmte Zeilennummer – dürfen Sie mit unserem Preprozessor benutzen.

Der Preprozessor bearbeitet Basic-Programme, die als ASCII-Datei vorliegen. Erzeugen können Sie solch eine Datei mit jedem Editor. Wordstar ist dazu genauso geeignet wie der Editor des Locomotiv- (beim CPC 6128, 664 und 464) und Mallard-Basic (beim Joyce). Beim Speichern auf Kassette oder Diskette müssen Sie nur an den SAVE-Befehl die Anweisung »,A« anhängen. Diese Maßnahme verwandelt nämlich alle Basic-Token wieder in ASCII-Zeichen und speichert das Programm als ASCII-Datei. Listing 1 zeigt als Beispiel ein unbehandeltes Programm, das mit einer REPEAT-UNTIL-Schleife und Marken arbeitet. Listing 2 zeigt die vom Preprozessor umgesetzte ablauffertige Routine.

In Listing 1 sehen Sie, daß die Marken mit »*@« beginnen und mit »@« enden. Beim Aufruf dieser Marken mit GOTO oder GOSUB muß zur Einleitung »@« statt »*@« stehen (also »GOTO @Marke@«). Die Schlüsselworte »REPEAT« und »UNTIL <Bedingung>« müssen immer alleine in einer Zeile stehen. Ansonsten dürfen Sie Ihr Basic wie gewohnt benutzen.

Listing 3 beinhaltet den eigentlichen Preprozessor. Unter dem Mallard-Basic des Joyce wird das Programm einfach eingegeben und mit RUN gestartet. Es empfängt Sie die Frage nach der zu lesenden und der zu schreibenden Datei. Die zu lesende liegt im ASCII-Format vor. Nachdem der Preprozessor fertig ist, liegt unter dem Namen des »Outfile« das fertige Programm im ASCII-Format auf dem Datenträger vor. Beim Laden wandelt sowohl der Mallard- wie auch der Locomotiv-Interpreter das Programm automatisch in eine lauffähige Version um.

Besitzer des CPC 6128, 664 und 464 müssen im Listing 3 die Zeilen aus Listing 4 ersetzen. Dann läuft auch bei ihnen das Programm korrekt. Da unser Preprozessor vollständig in Basic geschrieben ist, können Sie ihn jederzeit erweitern. Das Prinzip erkennt man leicht aus dem dokumentierten Listing. (A. Widmer/hg)

Programm-Steckbrief	
Name:	Preprozessor
Computer:	Joyce, CPC 464/664/6128
Datenträger:	Kassette/Diskette
Besonderes:	läuft nach entsprechenden Änderungen auf allen Schneider-Computern

```
REPEAT
  INPUT "ZAHL EINGEBEN: ",X%
  ON X% GOTO @MARKE_1@, @MARKE_2@, @C@, @C@, @LOTTI@
    GOSUB @PROMPT@: GOTO @LOOPEND@
*@MARKE_1@  PRINT "DIES IST MARKE 1":GOTO @LOOPEND@
*@MARKE_2@  PRINT "MARKE 2"         :GOTO @LOOPEND@
*@C@  REPEAT
    INPUT "ZAHL EINGEBEN: ",Y%
  UNTIL Y%=1 OR Y%=2
  GOTO @LOOPEND@
*@LOTTI@  INPUT "LOTTI .. ",SPRUCH$:IF SPRUCH$<>"I WOT DI" THEN @LOTTI@
*@LOOPEND@  REM --- LOOP-END
UNTIL X%=9
END
*@PROMPT@  PRINT "ZAHL IST GROESSER 5"
  RETURN
```

Listing 1. Unbehandelt ähnelt das Quell-Programm einem Pascal-Listing

```
 1 REM --- BEGIN OF REPEAT-LOOP
 2   INPUT "ZAHL EINGEBEN: ",X%
 3   ON X% GOTO 5 , 6 , 7 , 7 , 11
 4     GOSUB 15 : GOTO 12
 5   PRINT "DIES IST MARKE 1":GOTO 12
 6   PRINT "MARKE 2"         :GOTO 12
 7 REM --- BEGIN OF REPEAT-LOOP
 8     INPUT "ZAHL EINGEBEN: ",Y%
 9 IF NOT (Y%=1 OR Y%=2) THEN 7
10   GOTO 12
11   INPUT "LOTTI .. ",SPRUCH$:IF SPRUCH$<>"I WOT DI" THEN 11
12   REM --- LOOP-END
13 IF NOT (X%=9) THEN 1
14 END
15   PRINT "ZAHL IST GROESSER 5"
16   RETURN
```

Listing 2. Jetzt hat der Preprozessor aus dem »Fast-Pascal«-Programm eine waschechte Basic-Routine gemacht

```
10 REM
12 REM  PROGRAM MBASIC-PREPROCESSOR
14 REM
20 REM --- KONSTANTEN
22 TRUE%=-1:FALSE%=0
24 MAXINDEX%=500:MAXSTUFE%=20
30 REM --- VARIABLEN-FELDER
32 DIM PZEILE$(MAXINDEX%),SBEGINN%(MAXST
UFE%)
100 REM ------
102 REM  MAIN
104 REM ------
110 ON ERROR GOTO 3000
120 PRINT CHR$(27);CHR$(12)
122 PRINT"*** MBASIC - PROGRAM-PREPROCES
SOR (V1.1)":PRINT
124 LINE INPUT "INFILE  : ",INFILE$
126 LINE INPUT "OUTFILE : ",OUTFILE$
128 PRINT
130 REM --- PROGRAMMTEXT EINLESEN
132  GOSUB 1000
140 REM --- ZEILEN NUMMERIEREN
142  GOSUB 300
150 REM --- MARKEN TAUSCHEN
152  GOSUB 500
160 REM --- REPEAT .. UNTIL - SCHLAUFEN
TAUSCHEN
162  GOSUB 800
170 REM --- GEAENDERTEN PROGRAMMTEXT SPE
ICHERN
172  GOSUB 2000
180 PRINT CHR$(27);CHR$(12)
190  ON ERROR GOTO 0
200 END
300 REM
302 REM PROCEDURE  ZEILEN NUMMERIEREN
304 REM
310 PRINT"ZEILEN WERDEN NUMMERIERT.";
320 FOR ZNUM%=1 TO TEXTENDE%
330    PZEILE$(ZNUM%)=MID$(STR$(ZNUM%),2)
+" "+PZEILE$(ZNUM%)
340    PRINT".";
350 NEXT ZNUM%
360 PRINT
370 RETURN
500 REM
502 REM PROCEDURE  MARKEN TAUSCHEN
504 REM
510 PRINT"MARKEN WERDEN GESUCHT UND GETA
USCHT.";
520 FOR ZNUM%=1 TO TEXTENDE%
530    MPOS1%=INSTR(PZEILE$(ZNUM%),"*@")
540    IF MPOS1%=0 THEN 650
550       PRINT".";
560       MPOS2%=INSTR(MPOS1%+2,PZEILE$(ZN
UM%),"@")
570       MARKE$=MID$(PZEILE$(ZNUM%),MPOS1
%+1,MPOS2%-MPOS1%)
580       PZEILE$(ZNUM%)=LEFT$(PZEILE$(ZNU
M%),MPOS1%-1)+MID$(PZEILE$(ZNUM%),MPOS2%
+1)
590       FOR I%=1 TO TEXTENDE%
600          MPOS%=INSTR(PZEILE$(I%),MARKE$
)
610          IF MPOS%=0 THEN 640
620             PZEILE$(I%)=LEFT$(PZEILE$(I%
),MPOS%-1)+MID$(STR$(ZNUM%),2)+" "+MID$(
PZEILE$(I%),MPOS%+LEN(MARKE$))
630             GOTO 600
640       NEXT I%
650 NEXT ZNUM%
660 PRINT
670 RETURN
800 REM
802 REM PROCEDURE  REPEAT .. UNTIL-SCHLE
IFEN TAUSCHEN
804 REM
810 PRINT"REPEAT .. UNTIL - SCHLEIFEN WE
RDEN AUSGETAUSCHT.";
820 FOR ZNUM%=1 TO TEXTENDE%
830    IF INSTR(PZEILE$(ZNUM%),"REPEAT")=
0 THEN 870
840       PRINT".";
850       STUFE%=STUFE%+1:SBEGINN%(STUFE%)
=ZNUM%
860       PZEILE$(ZNUM%)=MID$(STR$(ZNUM%),
2)+" REM --- BEGIN OF REPEAT-LOOP"
870    UPOS%=INSTR(PZEILE$(ZNUM%),"UNTIL
")
880    IF UPOS%=0 THEN 920
890       PZEILE$(ZNUM%)=MID$(STR$(ZNUM%),
2)+" IF NOT ("+MID$(PZEILE$(ZNUM%),UPOS%
+6)+") THEN"+STR$(SBEGINN%(STUFE%))
900       STUFE%=STUFE%-1
920 NEXT ZNUM%
930 PRINT
940 RETURN
1000 REM
1002 REM PROCEDURE  PROGRAMMTEXT VON FIL
E EINLESEN
1004 REM
1010 PRINT"PROGRAMMTEXT WIRD EINGELESEN.
";
1020 OPEN INFILE$ AS 1 'FUER MALLARD-BAS
IC
1030 WHILE NOT EOF(1)
1040    TEXTENDE%=TEXTENDE%+1
1050    PRINT".";
1060    LINE INPUT#1,PZEILE$(TEXTENDE%)
1070 WEND
1080 PRINT
1090 CLOSE 1
1100 RETURN
2000 REM
2002 REM PROCEDURE  GEAENDERTEN PROGRAMM
TEXT AUFZEICHNEN
2004 REM
2010 PRINT"PROGRAMMTEXT WIRD AUFGEZEICHN
ET.";
2020 ' RESET
2030 OPEN OUTFILE$ AS 1
2040 FOR ZNUM%=1 TO TEXTENDE%
2050    PRINT".";
2060    PRINT#1,PZEILE$(ZNUM%)
2070 NEXT ZNUM%
2080 PRINT
2090 CLOSE 1
2100 RETURN
3000 REM
3002 REM PROCEDURE  FEHLER
3004 REM
3010 PRINT CHR$(27);CHR$(12)
3020 IF STUFE%<0 THEN FEHLER$="SCHLEIFEN
 NICHT KORREKT VERSCHACHTELT!":GOTO 3050
3022 IF STUFE%>MAXSTUFE% THEN FEHLER$="Z
U VIELE VERSCHACHTELUNGEN!":GOTO 3050
3030 IF TEXTENDE%>MAXINDEX% THEN FEHLER$
="PROGRAMM ZU LANG!":GOTO 3050
3040 FEHLER$="NUMMER "+STR$(ERR)+" IN ZE
ILE "+STR$(ERL)
3050 PRINT"*** FEHLER: "+FEHLER$+" - PRO
GRAMM ABGEBROCHEN"
3060 RESUME 190
```

Listing 3. Mit »Preprozessor« wird Ihr Basic noch leistungsstärker

```
120 CLS
180 CLS
1020 OPENIN INFILE$
1030 WHILE NOT EOF
1060    LINE INPUT#9,PZEILE$(TEXTENDE%)
1090 CLOSEIN
2020 REM
2030 OPENOUT OUTFILE$
2060    PRINT#9,PZEILE$(ZNUM%)
2090 CLOSEOUT
3010 CLS
```

Listing 4. Besitzer des CPC müssen nur wenige Zeilen ändern

Voll automatisiert

Auch das CP/M des Joyce bietet genügend Hilfsprogramme, die dem Benutzer die Bedienung vereinfachen.

Wenn man viel mit Compilern arbeitet oder oft gleiche Programmabläufe benötigt, sparen SUBMIT-Dateien viel Arbeit. Solche Dateien kann man entweder mit Locoscript (ASCII-File) oder mit dem Texteditor »RPED« unter Basic anlegen. Um sich den Aufbau einer solchen Datei klarzumachen, schauen Sie sich die Datei »PROFILE« auf der Systemdiskette an. Der erste Befehl (»SETDEF«) bewirkt, daß CP/M erst nach einem SUB-File sucht und dies aufruft, wenn sich die CP/M-Datei SUBMIT.COM auf der Diskette befindet. Findet CP/M kein SUB-File mit dem angegebenen Namen, sucht es nach einem COM-File. Das System beginnt die Suche beim M- und setzt sie im A-Laufwerk fort. PIP kopiert dann ein paar wichtige CP/M-Files in die RAM-Floppy. Um andere Files zu übertragen, geben Sie folgendes ein:

»<Laufwerk x:=Laufwerk y:Programmname«. Um PIP zu verlassen, brauchen Sie nur »<« einzugeben. Wenn Sie beispielsweise für den C-Basic-Compiler eine SUB-Datei anlegen wollen, die das File »prg.bas« compiliert, linkt und schließlich startet, tippen Sie folgendes in eine ASCII-Datei ein:

```
CB80 prg
LK80 prg
prg
```

Das wäre alles. Der Name der Datei muß die Extension .SUB als Idendifikat enthalten Das File »SUBMIT.COM« muß entweder im Laufwerk M oder A stehen (SETDEF beachten). Eine Submitdatei läßt sich wie ein normales CP/M-Programm starten, also mit »programmname« und ENTER.

(Michael Straßer/ja)

Reset perfekt

Vielleicht haben Sie sich auch schon einmal gewundert, warum der Reset des Joyce nicht immer den gewünschten Erfolg bringt. Hier ein Trick, der immer funktioniert.

Nach dem Anschalten bootet der Joyce die eingelegte Diskette automatisch (Autostart). Möchte man aus einem Programm (oder aus CP/M) eine Diskette booten, so geschieht dies durch Einlegen der betreffenden Diskette und gleichzeitiges Drücken der Tasten SHIFT, EXTRA und EXIT. Hat man jedoch eine falsche Diskette (ohne Systemprogramm) erwischt, quittiert Joyce dies mit dreimaligem Piepsen und stellt seinen Dienst vorübergehend ein. Bei einem erneuten Versuch, durch Drücken der drei Tasten die Diskette nochmals zu booten, passiert nichts. Es ist jedoch nicht erforderlich, den Computer für einen weiteren Versuch auszuschalten. Es genügt, das Drücken der drei Tasten über die Leertaste zu bestätigen. (Jörg Höhnerbach/ja)

Na Logo!

Außer dem Mallard-Basic gehört zum Lieferumfang des PCW 8256/8512 die Programmiersprache Logo. Aber wer sich damit auseinandersetzt, steht bald vor »unlösbaren« Problemen.

Einsteiger, die mit Logo erstmals konfrontiert sind, werden sich schwertun, beim Aufbau von Listen in Prozeduren, die richtigen Klammerausdrücke zu finden. Das Benutzerhandbuch zeigt folgendes Beispiel:

```
repeat 4 [fd 60 rt 90]
```

Bei dem Versuch, diesen Befehl auszuführen, reagiert Logo mit wenig Verständnis für das Eingabeformat. Obwohl die Klammerzeichen bei der Tastaturbelegung für Logo verfügbar sind, werden sie nicht entsprechend interpretiert. Das hat folgenden Grund: Bei der deutschen Tastaturbelegung muß man anstelle der eckigen Klammern die Buchstaben Ä und Ü benutzen. Obige Zeile muß also wie folgt aussehen:

```
repeat 4 Äfd 60 rt 90Ü
```

Nach einiger Zeit gewöhnt man sich an das merkwürdige Aussehen der Befehlszeilen und kann dann vernünftig mit Logo arbeiten. Wer sich näher für Logo interessiert, dem sei das 2. Schneider Sonderheft von Happy-Computer (SH1/86) empfohlen. Dort gibt es eine Einführung in diese interessante Sprache. (Jörg Höhnerbach/ja)

Buchstabensalat

Man programmiert in Mallard-Basic und hat irgendwann den ganzen Bildschirm voller »Müll«. Wie Sie ihn wieder loswerden, steht aber nirgends im Handbuch – genausowenig, wie die Fähigkeit des Joyce, Hardcopies anzufertigen.

Viele Benutzer des Joyce haben sicher schon vergeblich nach zwei bestimmten Befehlen in dem umfangreichen Benutzerhandbuch gesucht. Sie dürfen Ihre Suche beenden: Mallard-Basic kennt keinen Befehl zum Bildschirmlöschen!

Diese Funktion läßt sich aber durch die Eingabe der nachfolgenden Zeilen erzielen.

```
10 cl$=(chr$(27)+"E")+(chr$(27)+"H")
20 print cl$
```

Hat man die notwendigen Steuerzeichen in einer Programmzeile definiert (hier Zeile 10), löscht man den Bildschirm durch den Befehl in Zeile 20. Es empfiehlt sich, die zwei Zeilen zu speichern und vor der Eingabe jedes neuen Programms in den Speicher zu laden.

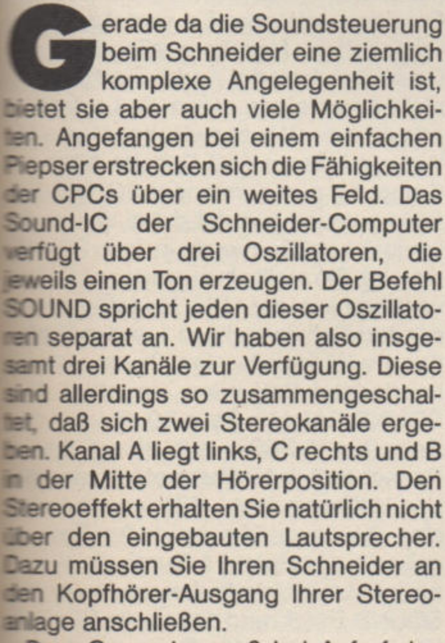
In vielen Situationen ist es wünschenswert, eine Hardcopy auszudrucken. Dies bedeutet eine Ausgabe des aktuellen Bildschirminhaltes Punkt für Punkt auf den Drucker.

Durch gleichzeitiges Drücken der Tasten EXTRA und PTR ist dies jederzeit möglich. (Jörg Höhnerbach/ja)

Der Ton macht die Musik



Das SOUND-Kommando, das beim CPC die Klangerzeugung steuert, ist einer der komplexesten Befehle dieses Computers. Wie arbeitet der Befehl, und welche Klangeffekte sind machbar? Anhand eines Beispiels erkennen Sie die Zusammenhänge.



Gerade da die Soundsteuerung beim Schneider eine ziemlich komplexe Angelegenheit ist, bietet sie aber auch viele Möglichkeiten. Angefangen bei einem einfachen Piepser erstrecken sich die Fähigkeiten der CPCs über ein weites Feld. Das Sound-IC der Schneider-Computer verfügt über drei Oszillatoren, die jeweils einen Ton erzeugen. Der Befehl SOUND spricht jeden dieser Oszillatoren separat an. Wir haben also insgesamt drei Kanäle zur Verfügung. Diese sind allerdings so zusammengeschaltet, daß sich zwei Stereokanäle ergeben. Kanal A liegt links, C rechts und B in der Mitte der Hörerposition. Den Stereoeffekt erhalten Sie natürlich nicht über den eingebauten Lautsprecher. Dazu müssen Sie Ihren Schneider an den Kopfhörer-Ausgang Ihrer Stereoanlage anschließen.

Dem Computer muß bei Aufruf des SOUND-Befehls bekannt sein, auf welchen Kanal er den Ton ausgeben soll. Als erste Angabe hinter dem Befehl steht daher die Kanalnummer. Der Logik zufolge sollten nun die Kanäle mit den Nummern 1, 2 und 3 bezeichnet sein. Das ist aber nicht der Fall: Kanal A bekommt zwar eine »1« und Kanal B eine »2«. Wenn jedoch eine Ausgabe an Kanal C gehen soll, ist eine »4« einzugeben. Dies hat etwas mit der binären Codierung der Kanäle zu tun. Für uns ergibt sich daraus eine recht praktische Anwendung. Wenn wir nämlich zwei oder alle drei Kanalnummern addieren, erhalten wir als Ergebnis einen Klang auf allen angesprochenen Kanälen gleichzeitig. Beispielsweise spricht ein mit »SOUND 3,...« produzierter Ton sowohl Kanal A als auch Kanal B an.

Wie kommt nun die Tonhöhe, also die Frequenz eines Tones, zustande? Auch wenn wir die genaue Frequenz kennen, ist etwas Rechenarbeit nötig, um den richtigen Wert zu ermitteln. Der Computer erwartet die Tonhöhe nämlich in codierter Form. Nehmen wir an, wir wollten den Kammerton A ausgeben.

Töne mit »Hertz«

Dazu benötigen wir eine Frequenz von 440 Hertz (Hertz ist die physikalische Einheit für Schwingungen pro Sekunde). Jetzt ist es aber nicht möglich, diese »440« als zweiten Parameter einzugeben, da der Computer mit anderen Frequenzbereichen arbeitet. Er errechnet nämlich die Frequenz aus seinem internen Systemtakt. Aus dieser Grundschwingung von 62500 Hertz gewinnt er durch Teilung alle anderen benötigten Frequenzen.

Um den richtigen Wert zu ermitteln, dividieren Sie also 62500 durch die gewünschte Frequenz (440 Hertz). So erhalten Sie die Zahl 142. Diese stellen Sie nun an die zweite Stelle im SOUND-Kommando. Nun ist es relativ mühsam, für jeden Ton die Tonhöhe nach diesem Prinzip zu bestimmen. Gott sei Dank hilft Ihnen die Tabelle im Anhang des Bedienerhandbuches Ihres Computers weiter. Dort finden Sie zu jeder Note in den gängigen Oktaven die notwendigen Parameter. Wenn Sie jetzt also auf Kanal A den Kammerton A ausgeben wollen, lautet das SOUND-Kommando bis jetzt:

```
SOUND 1,142
```

Als dritten Parameter fordert der SOUND-Befehl noch die Eingabe über die Dauer des Tones einzugeben und zwar in hundertstel Sekunden. Wenn Sie hier also »100« eingeben, beträgt die Tondauer eine ganze Sekunde. Da in der Regel meist aber die Notenwerte der Töne bekannt sind, empfiehlt sich, die Werte für bestimmte Notenlängen einmalig festzulegen und dann die entsprechenden Werte für andere Töne danach zu berechnen.

Die bis jetzt beschriebenen Parameter sind Pflichtübungen bei der Eingabe jedes SOUND-Kommandos. Kommen wir jetzt zur Kür. Parameter Nummer 4 regelt die Lautstärke: Sie können dabei

zwischen Werten von 0 (stumm) bis 15 (volle Lautstärke) wählen. Sie dürfen diese Angabe aber auch weglassen.

Auch variable Lautstärkenveränderungen im Tonverlauf sind machbar. Dazu verhilft der nächste Parameter. Er bestimmt, welches der 15 ENV-Register (ENV steht für envelope = umhüllen) für diesen Ton gilt. Sie können also ein An- und Abschwellen eines Tones in der Lautstärke definieren. Damit ist beispielsweise ein Vibrato-Effekt realisierbar. Gerade mit »ENV« lohnt es sich, ein bißchen herumzuexperimentieren; die Variationen sind sehr vielfältig in ihren klanglichen Auswirkungen.

Das ENV-Kommando hat folgende Struktur: An erster Stelle steht die Nummer der Lautstärkehüllkurve. Es folgen maximal fünf Gruppen mit je drei Werten, wobei im ersten Wert die Anzahl der Wiederholungen, im zweiten die Höhe der Volumenänderung bei jeder Wiederholung – und zwar positiv oder negativ – und im dritten Wert der Abstand zwischen den einzelnen Wiederholungen, also die Länge der Pause, anzugeben ist. Wenn Sie beispielsweise definieren: »ENV 1,15,-1,10,15,1,10«, bedeutet dies, daß sich zuerst 15mal die Lautstärke um den Betrag 1 vermindert, bevor sie sich dann in 15 weiteren Stufen wieder erhöht. Zwischen jeder Änderung wird eine Pause von der Länge einer zehntel Sekunde (10 x 0,01 Sekunden) eingelegt.

Laut und leise

ENV wirkt aber nicht allein, sondern dient nur als Unterprogramm zum SOUND-Befehl. Denn schließlich existieren im ENV-Kommando ja keine Angaben über Tonhöhe oder Länge. Diese entnimmt der CPC bei der Klangausgabe dem SOUND-Befehl. Wenn Sie nun also »SOUND 1,142,300,15,1« eingeben, erklingt auf Kanal A der Kammerton »A« mit einer Länge von drei Sekunden in der maximalen Lautstärke (Lautstärke 15) und Lautstärkefolge (ENV) 1. Tippen Sie jetzt einmal die beiden Kommandos ein und hören Sie sich das Ergebnis an. Der Lautstärkewert 15 stellt die Ausgangsbasis dar, ab der dann die Änderungen durch ENV wirken.

Nun läßt sich nicht nur die Lautstärke, sondern auch die Tonhöhe nach diesem Prinzip verändern. Damit kann man zum Beispiel täuschend echt das Heulen einer Sirene imitieren. Aber auch musikalisch ist der Effekt nützlich, denn verschiedene Töne lassen sich fließend »ineinander ziehen«. Und ein Tremolo ist überhaupt kein Problem. Dazu dient die Tonänderungsfolge »ENT« (envelope tone). Auch hier stehen Ihnen 15 Register zur Verfügung, die Sie variieren können. Die Syntax ist ähnlich zu »ENV«. Bei »ENT« gibt der erste Wert wiederum das Register an – beziehungsweise die Nummer der Tonfolge, mit der gearbeitet werden soll – der zweite Wert die Höhe der Frequenzverschiebung, der dritte die Anzahl der Wiederholungen und der vierte die Pausen zwischen den einzelnen Wiederholungen.

Die Tonänderungsfolge rufen Sie auf, indem Sie die betreffende Nummer an sechster Stelle im SOUND-Befehl angeben. Bevor wir uns mit der genauen Wirkung dieses Zusatzbefehles anhand eines Beispiels befassen, soll nun noch kurz der letzte Parameter beschrieben werden.

Die letzte Zahl hinter dem SOUND-Befehl sagt aus, ob und wie ein Rauscheffekt auftreten soll. Steht an dieser Stelle eine Null, unterbleibt das Rauschen; befindet sich hier eine Zahl zwischen 1 und 15, wird ein Rauschen unterschiedlicher Art zugefügt. Es handelt sich dabei jedesmal um sogenanntes »weißes Rauschen«, das je nach Eingabe einer Zahl dumpfer oder heller klingt. Dieser Effekt ist schwer zu beschreiben, deshalb sollten Sie ihn einmal ausprobieren und alle Zahlen durchspielen. So sehen Sie am besten, welcher Effekt bei jeder dieser Zahlen auftritt. Eine FOR-NEXT-Schleife kann Ihnen dies abnehmen, indem Sie zum Beispiel folgendes eingeben:

```
10 FOR i=1 TO 15
20 SOUND 1,100,100,10,0,0,i
30 NEXT i
```

Nach dem Start dieses kleinen Programms hören Sie einen Ton und dann nacheinander die verschiedenen Möglichkeiten des Rauschens. Wollen Sie das Rauschen allein hören, geben Sie anstelle der ersten »100« eine »0« ein. Zur besseren Identifikation der Rauschklänge können Sie sich innerhalb der Schleife den aktuellen Rauschwert mit »PRINT i« ausgeben lassen.

Nachdem Sie nun die Syntax des SOUND-Kommandos kennen und auch die Wirkung eines zugeschalteten »ENV« oder »ENT« im Griff haben, sollten Sie eigentlich in der Lage sein, ein kleines Programm mit diesen Befehlen zu schreiben. Programmieren wir jetzt einen kleinen Sound-Gag, der einen startenden und wegfliegenden Hubschrauber simuliert. Ein startender Hubschrauber läßt zunächst einen langsam laufenden Motor hören, der dann immer mehr beschleunigt, bis er startet, in der Ferne verschwindet. Wir versuchen also dies auch mit Hilfe der SOUND-Kommandos nachzuvollziehen.

```
10 ENT 1,200,-2,10
20 SOUND 1,500,2000,15,1,1,3
30 ENV 2,1,-1,12
40 FOR i=14.9 TO 0 STEP -1
50 SOUND 1,250,10,INT(i)+1,2,5
60 SOUND 1,0,2
70 NEXT i
```

Wir wollen hier einmal zwei Phasen realisieren. Zunächst das Starten. Der Motor läuft langsam hoch. Dann fliegt der Hubschrauber und entfernt sich langsam. Beide Geräusche müssen Sie daher mit relativ komplexen Lautstärke- und Tonfolgen programmieren. Den ersten Teil verwirklicht ENT und ein konstant zugeschaltetes Rauschen. Der zweite Teil bedarf der Definition einer Lautstärkehüllkurve (entfernen durch leiser werden) und einer gleichzeitigen, unterbrochenen Ausgabe des Tones, um die Rotorgeräusche zu erzeugen. Mit Hilfe der ENT-Folge in Zeile 10 erhöhen Sie in 200 Schritten langsam die Tonhöhe. Das zugemischte Rauschen in Zeile 20 (letzter Parameter) sorgt für den maschinenhaften Klang.

Auf und davon

Das Fliegen der Maschine stellt sich als eine relativ trickreiche Angelegenheit heraus. Zum einen vermindern Sie ganz langsam in einer FOR-NEXT-Schleife die Lautstärke. Die Schrittweite ist der Einfachheit halber gering gewählt. Die Änderungen der Lautstärke sollen dabei in möglichst kleinen Schritten erfolgen. Es sind 150 dazu vorgesehen. Nun kann allerdings der Gesamtumfang der Änderung 15 Werte nicht übersteigen, denn die Lautstärke erlaubt ja nur Werte zwischen 0 und 15. Während dieser 150 Schritte führt der Computer nun permanent SOUND-Befehle aus und benutzt dabei die zweite Lautstärkehüllkurve, die in Zeile 30 definiert ist. Sie stellt ein nur kurzes Hüllkurvenfragment dar. Die Lautstärke vermindert sich danach um 1 und anschließend wird jeweils eine Pause von 12 hundertstel Sekunden eingefügt. Der SOUND-Befehl in Zeile 50 gibt nun den Grundton (»250«) mit einem hellen Rauschen (letzter Parameter) unter dieser Hüllkurve aus. »ENV« vermindert also die Lautstärke bei jeder Tonausgabe um den Betrag 1. Durch die gleichzeitige Lautstärkeänderung im ersten SOUND-Befehl und das zweite SOUND-Kommando, das für Ruhe sorgt, entsteht ein durchbrochener Klang, der einem Hubschrauberknattern sehr ähnlich ist.

Die angegebenen Parameter ergaben sich aus einigen Probeläufen. Es fällt Ihnen aber nun sicher nicht mehr schwer, zu noch besseren Ergebnissen zu gelangen.

Probieren Sie es doch einmal aus und ... einen guten Flug!

(Carsten Straush/ja)

Farb-Spielereien auf dem Schneider

Obwohl Computer in erster Linie für ernsthafte Anwendungen entwickelt wurden, lernt man sie am besten spielerisch kennen. Hier finden Sie einige Anregungen aus dem Bereich der Grafik.

Das Staunen ist oft groß, wenn man sieht, was sich Softwarefirmen einfallen lassen. Bei einigen Softwaretricks fragt man sich: Wie ist das möglich? Wie gelangt man an die Leistungsgrenze seines Computers? Ganz einfach: Spielerisch und mit System.

Beim Schneider bieten sich zwei sehr interessante Arbeitsfelder an. Die Kombination von Text und Grafik durch den hochauflösenden Grafikspeicher und die Farbgebung. Besonders letzterer bietet durch zahlreiche Varianten ungeahnte Ergebnisse. Beginnen wir bei den Grundlagen. Die Farbe wird beim CPC indirekt definiert. Es ist also nicht festgelegt, daß 6 beispielsweise rot sein soll und damit bei »PEN 6« die Schriftfarbe auf rot umgeschaltet wird, wie dies bei anderen Computern der Fall ist. Vielmehr sind zwischen die Farbnummer und die dann gesetzte Schrift- beziehungsweise Hintergrundfarbe eine Reihe von Farbregistern, die sogenannten INK-Register, eingeschaltet. »PEN 6« greift also auf den Inhalt des Farbregisters 6 und die Farbe, die dort steht, zurück. Hierbei ist allerdings auch noch eines zu beachten.

Der Schneider verfügt bekanntermaßen über drei verschiedene Darstellungsmodi, Modus 0 (mit 20 Zeichen pro Zeile), Modus 1 (mit 40 Zeichen pro Zeile) und Modus 2 (den hochauflösenden Modus mit 80 Zeichen). Nur bei Modus 0 wäre die oben angegebene Farbanweisung von Erfolg gekrönt. Der CPC wendet nämlich auf alle Farbwerte die Modulo-Funktion an. Im Modus 0 werden alle Farbwerte MOD 16 genommen, im Modus 1 mit MOD 4 und im Modus 2 mit MOD 2. Machen wir uns an einem Beispiel klar, was hier passiert. Die Modulo-Funktion bewirkt eine Division durch den nachfolgenden Wert, wobei der Rest als Funktionsergebnis übergeben wird. »11 MOD 4« ergibt beispielsweise den Wert 3. Damit verstehen Sie nun auch, wieso der CPC trotz verschiedener Farbangaben manchmal mit derselben Farbe schreibt. Es ist egal, ob Sie im Bildschirmmodus 1 PEN 1, PEN 5, PEN 9 oder PEN 13 eingeben. In jedem Fall wird als Schriftfarbe der Farbwert genommen, der in Register 1 gespeichert ist. Alle diese Werte haben nämlich als Rückgabe der Modulo-Funktion das Ergebnis 1. Probieren Sie es aus! Im Bildschirm-Modus 2 kehrt ein Farbwert sogar alle zwei Farbangaben wieder. PEN 1 entspricht also PEN 3, PEN 5 etc. Daraus resultiert auch eine Unzulänglichkeit des Computers, die den Anfänger oft zur Verzweiflung treibt. Wenn Sie im Modus 2 die Pen-Farbe 2 auswählen, so hängt sich der Schneider scheinbar auf. Kein ESC kann ihn in die Welt der Lebenden zurückbringen. Da aber 2 MOD 2 gleich 0 ist, hat der CPC für die Vordergrundfarbe dieselbe Farbe wie für den Hintergrund ausgewählt.

Der schwarze Schirm

Nun wissen wir, wie die Farbgebung beim Schneider funktioniert. Diese Art der Farbdefinition bietet eine ganze Reihe von Effekten, die man mit einer direkten Farbdefinition nicht erreichen könnte. Erstes Beispiel: Das Auftasten von Farben. Dieser Trick bietet die Lösung für ein Problem, das des öfteren bei der Titelgestaltung auftaucht. Die Titelausgabe in Basic braucht meist einige Zeit. Nun sieht es aber nicht besonders professionell aus, wenn man ein Titelbild oder einen Spielschirm bei Action-Spielen Stück für Stück aufbaut. Eine Lösung dieses Problems ist mit Hilfe der INK-Register möglich. Man setzt zunächst alle »INKs« auf 0, also schwarz. Dann schreibt man den Bildschirmtitel wie gewohnt. Als dritter Schritt werden dann die einzelnen Farbregister mit den darzustellenden Farben belegt. Ergebnis: Zunächst bleibt der Schirm dunkel. Die Darstellung schwarz auf schwarz sieht man ja nicht. Nach einer Kunstpause wechselt der Schirm seine Farbe und auf einen Schlag erscheint das Titelbild. Diesen Vorgang zeigt Ihnen das folgende Listing:

```
10 INK  0,0:INK 1,0:INK 2,0:
   INK 3,0
20 PEN 1:PRINT"DIES IST MIT
   PEN 1 GESCHRIEBEN"
30 PEN 2:PRINT"UND DIES MIT
   PEN 2"
40 PEN 3:PRINT"JETZT NOCH EINE
   ZEILE IN PEN 3"
50 FOR i=1 TO 200:NEXT
60 INK 1,24:INK 2,6:INK 3,11
```

Dieselbe Technik können Sie natürlich auch verwenden, um in einem erklärenden Text Worte zu unterstreichen. Tippen Sie dazu einmal

```
10 INK 0,0:INK 1,24:INK 2,0:PAPER
   0:PEN 1
20 PRINT"In einem ganz normalen
   Erklaerungstext ";:PAPER
   2:PRINT"soll diese Stelle
   ";:PAPER 0:PRINT"besonders
   hervorgehoben werden."
30 PRINT"So sieht der Text im
   Normalfall aus"
40 FOR i=1 TO 300:NEXT
50 INK 2,6
60 PRINT"Und jetzt mit der
   hervorgehobenen Stelle!"
```

ein.

Es geht sogar noch kürzer. Mit einem Trick können Sie innerhalb eines Zeichens die Farbe umschalten. Wie Sie vielleicht wissen, verfügt der CPC über einen sogenannten Transparent-Modus. In dieser Darstellungsart werden die Zeichen übereinander geschrieben. Der CPC legt jedes Zeichen als hochauflösendes Grafiksymbol in einer Matrix aus 8 mal 8 Bildpunkten im Grafikspeicher ab. Für ein A wird kein Code gespeichert, sondern die Bildpunkte, die dem Buchstaben entsprechen, im Grafikspeicher mit der Schriftfarbe gesetzt. Analog dazu wird der durch das Zeichen nicht belegte freie Raum mit der Hintergrundfarbe belegt. Diesen letzten Teil der Operation, das Setzen der Hintergrundfarbe, können Sie durch Einschalten des Transparent-Modus verhindern. Es werden dann nur noch die durch das Zeichen selbst belegten Bildpunkte gesetzt. Die Umschaltung zwischen Normal- und Transparent-Modus geschieht dabei mit »PRINT CHR$(22)+CHR$(1)«. »PRINT CHR$(22)+CHR$(0)« schaltet wieder zurück. Probieren Sie einmal, nach der Eingabe von »CHR$(22)« den Bildschirm mit DEL zu löschen. Es funktioniert nicht mehr. Der Grund dafür ist einfach. DEL schreibt die zu löschende Zeichenposition mit einem Quadrat von der Größe eines Zeichens mit Hintergrundfarbe voll. Und genau dieses Beschreiben mit der Hintergrundfarbe funktioniert ja nicht mehr.



Schauen Sie sich einmal an, wie der Transparent-Modus wirkt. Ein einfaches PRINT-Kommando genügt dazu: »PRINT CHR$(22)«. Wenn Sie »PRINT"AAAAAA<6 mal CTRL-H> <6 mal SHIFT 0>"« eintippen, werden die beiden Zeichen überlagert. Sie erhalten unterstrichene Buchstaben. Auf diese Weise können Sie sich einen neuen Zeichensatz definieren. Daneben ist es natürlich möglich, in einem Zeichen mit mehr als zwei Farben zu arbeiten. Man überlagert einfach die dritte und jede folgende Farbe. Wenn Sie beispielsweise zwischen »AAA« und dem Strich die Farbe umschalten, wird der Strich in einer anderen Farbe dargestellt. Die gelben »As« werden rot unterstrichen.

Unterstreichen in Basic

```
10 INK 0,0:INK 1,24:INK 2,6
20 PEN 1:PAPER 0:LOCATE 1,10
30 PRINT"AAAAAA"
40 PEN 2:LOCATE 1,10
50 PRINT"<6 mal SHIFT 0>"
```

Wie können wir aber diese Eigenschaft des CPC für unsere Farbumschaltung innerhalb eines Zeichens benutzen? Stellen Sie sich vor, Sie überschreiben ein Zeichen mit einem neuen, wobei die Schriftfarbe des neuen Symbols mit der Hintergrundfarbe des alten übereinstimmt. Die Schriftfarbe des neuen Zeichens entnehmen Sie dabei einem neuen Farb-Register. Sie setzen also mehrere INKs auf denselben Wert. Bleiben Sie dabei im Normal-Modus, so wird nur das neue Zeichen dargestellt. Beim Transparent-Modus hingegen bleibt der Rest des alten Zeichens erhalten, der nicht von dem neuen überdeckt wird. Nach der Ausgabe sehen Sie zunächst das alte Zeichen, soweit es nicht überschrieben wurde. Durch Umdefinition des Farb-Registers, aus dem die Schriftfarbe kam, können Sie dann das neue überlagerte Zeichen einblenden.

Ein Beispiel soll dies illustrieren. Wenn Sie im Anhang Ihres Benutzerhandbuches nachschlagen, finden Sie dort als Zeichen 239 ein Raumschiff-ähnliches Gebilde. In der untersten Bildpunktreihe sind zwei Öffnungen ausgespart, die man mit ein wenig Phantasie als Triebwerke definieren kann. Allerdings scheint sich die Rakete in Ruhestellung zu befinden. Wir wollen nun dieses Manko beheben. Dazu definieren wir mit »Symbol« ein neues Zeichen. »SYMBOL 240,0,0,0,0,0,0,0, &x00101000« belegt die gesamte Fläche des Zeichens mit der Hintergrundfarbe. Nur zwei Punkte sind davon ausgenommen, die Triebwerköffnungen. Wir wollen nun dieses neue Zeichen und Symbol Nummer 239 übereinanderdrucken. Zuerst wird dabei Zeichen 239 ausgegeben. Die Überlagerung durch das Triebwerkfeuer folgt dann im zweiten Schritt.

```
10 MODE 0:INK 0,0:INK 1,11:INK
   2,0:PAPER 0:PEN 1
20 PRINT CHR$(22)
30 SYMBOL
   240,0,0,0,0,0,0,0,&<00101000
40 CLS:LOCATE 20,10:PRINT CHR$
   (239)
50 PEN 2:LOCATE 20,10:PRINT
   CHR$(240)
60 FOR i=1 TO 800:NEXT
70 INK 2,6,24
```

Durch Setzen des Farbregisters 2 auf 0 ist die Vordergrundfarbe von Zeichen 240 gleich der Hintergrundfarbe von Zeichen 239. Erst die Umdefinition der Farbe in Programmzeile 70 sorgt dafür, daß das Triebwerk gestartet wird. Es wird durch INK 2,0 wieder abgeschaltet. Man kann diesen Effekt allerdings auch in der umgekehrten Richtung einsetzen, um beispielsweise eine Öffnung zu schaffen, wo vorher keine war. Dies ist zum Beispiel bei Adventures oft der Fall. Durch eine Eingabe soll sich eine Tür im bereits dargestellten Spielbildschirm öffnen. Mit der Farbumdefinition ist dies leicht zu lösen. Man definiert wiederum zwei »INKs« auf dieselbe Farbe. Mit der einen »INK« wird dann die Tür, mit der anderen die Wand beschrieben. Ein einfaches INK-Kommando reicht aus, um auf Knopfdruck den Spielbildschirm zu verändern. »Elektro Freddy« arbeitet beispielsweise mit diesem Trick.

Man kann die Farbumdefinition allerdings nicht nur bei der Zeichenausgabe, sondern auch im Bereich der hochauflösenden Grafik einsetzen. Besonders wichtig ist dies, wenn man mit dem TAG-Befehl arbeitet. Diese Anweisung erlaubt, die Zeichenausgabe an der Stelle zu positionieren, an der sich der Grafikcursor befindet. Dabei stimmt die oberste linke Ecke des Ausgabezeichens mit der Position des Grafikcursors überein. So kann man ein Objekt von der Größe eines Zeichens kontinuierlich über den Bildschirm bewegen. Wir wollen dies einmal anhand unserer Rakete demonstrieren.

```
10 CLS:FOR i=600 TO 1 STEP-2:TAG:
   CALL&BD19:MOVE i,250:
   PRINT CHR$(239);:NEXT
```

Mit dieser Zeile wird das Raumschiff-Zeichen über den Bildschirm bewegt. Der MOVER-Befehl bewegt den Grafikcursor innerhalb der FOR-TO-Schleife langsam nach links und gibt durch den nachfolgenden PRINT-Befehl an dieser Stelle das Raumschiff-Zeichen aus. Das zwischengeschaltete CALL-Kommando hat dabei eine Synchronisationsfunktion. Der Bildschirm wird im Monitor dadurch aufgebaut, daß ein Elektronenstrahl über die gesamte Bildschirmfläche gelenkt wird, wobei er je nach Auslenkungswinkel und Auftastung die einzelnen Farbpunkte auf der Bildschirmoberfläche zum Aufleuchten bringt. Der Strahl wandert dabei zeilenweise in jeder fünfzigstel Sekunde einmal von oben links nach unten rechts. Danach wird der Strahl ausgetastet und läuft »blind« zu seinem Ausgangspunkt zurück. Ein Zeichen wird beim CPC Bildpunkt für Bildpunkt im Grafikspeicher abgelegt. Nun kann es vorkommen, daß der Strahl gerade in dem Moment Bildschirmpunkte darstellen will, in dem diese vom Computer farblich geändert werden. Ergebnis: Der eine Teil des Zeichens wird noch in der alten Position dargestellt; der Rest aber schon in der neuen. Unser Zeichen wird also auseinandergerissen. Wenn Sie den CALL-Befehl weglassen, werden Sie sehen, daß das Oberteil der Rakete etwas verschoben gegen das Unterteil dargestellt wird. Der CALL-Aufruf vermeidet diese Probleme. Er hält die nachfolgenden Operationen solange an, bis der Strahl zurückläuft.

Kehren wir zurück zur Farbumdefinition im Bereich der hochauflösenden Grafik. Obwohl Sie mit TAG ein Zeichen in Abhängigkeit von der Grafikcursorposition bewegen können, wird bei der HiRes-Grafik auch in diesem Fall die Farbe des Grafikcursors zum Setzen der einzelnen Bildpunkte verwendet. Es hat daher keinen Sinn, wenn Sie versuchen, mit »PEN« und »PAPER« irgendwelche Effekte zu erzielen. Da der Transparent-Modus in Zusammenhang mit »TAG« nicht funktioniert, ist dies auch kein Manko. Das Aufleuchten des Triebwerkfeuers können Sie auch durch ein »PLOT« realisieren. Dazu müssen wir uns allerdings ein wenig mit der Bildschirm-Arithmetik beschäftigen.

3,2,1 – PLOT

Wenn Sie mit x,y die Position bezeichnen, an der sich der aktuelle Grafikcursor befindet, so können Sie die Position der Triebwerkdüsen relativ einfach bestimmen.

Im Modus 0 werden jeweils vier Bildpunkte in der Horizontalen mit derselben Farbe belegt. Unsere Zeichenmatrix wird also in diesem Bildschirm-Modus auf je vier Bildschirmpunkte in der Horizontalen gedehnt. Je vier nebeneinanderliegende Bildpunkte erhalten denselben Farbwert. Bei unserer Rakete sind die ersten beiden Punkte unseres Zeichens in der unter-

sten Bildschirmlinie gesetzt (Vergleichen Sie dazu noch einmal die Zeichenmatrix im Anhang Ihres Handbuchs). Die Bildpunkte X bis X+7 in dieser Bildschirmlinie werden also bei der Ausgabe dieses Zeichen gesetzt. Wenn Sie nun 8 zu unserem X-Wert addieren, so treffen Sie genau auf den durch das Triebwerk freien dritten Punkt von links.

8 Punkte entfernt

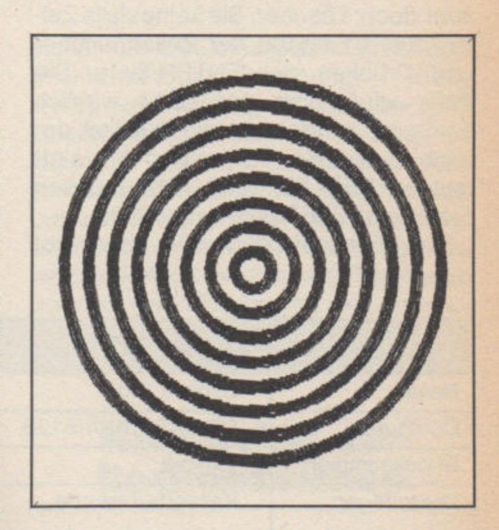
Die zweite Triebwerköffnung liegt dann wieder acht Bildpunkte weiter in der Position X+16. Nach der X-Verschiebung müssen Sie nun noch die Y-Abweichung bestimmen. Für jede Bildschirmlinie, die Sie weiter nach unten wollen, müssen Sie eine 2 von der Y-Koordinate subtrahieren und umgekehrt. Sie erhalten damit Y-14 als Vertikalwert. Der Grafikcursor liegt in der obersten Bildschirmlinie dieses Zeichens und sieben Bildschirmlinien tiefer spielt sich unser Triebwerkfeuer ab. Nach dieser Vorarbeit ist das Setzen der Triebwerkpunkte kein Problem mehr. Wir müssen nur die Punkte (X+8,Y-14) und (x+16,Y-14) mit einer geeigneten Farbe setzen. Wenn Sie das Raumschiff auf die Position 300,200 setzen, so sind im Modus 0 die Punkte (308,186) und (316,186) zu setzen. Die notwendigen Befehle dazu:

```
10 INK 0,0:INK 1,11:INK 2,0:MODE
   0:PLOT 300,200,1:TAG:
   PRINT CHR$(239);
20 PLOT 308,186,2:PLOT 316,186,2
30 FOR i=1 TO 800:NEXT
40 INK 2,24,6
```

Dabei werden die Punkte zuerst mit der auf 0 gesetzten Farbe 2 geschrieben. Erst nach der Zeitverzögerung in Zeile 30 wird dann das Triebwerk zugeschaltet. Zwei Anmerkungen sind in diesem Zusammenhang noch notwendig. Erstens muß nach jeder Zeichenausgabe mit TAG ein Semikolon stehen. Sonst wird als Äquivalent zum RETURN-Befehl ein linksgerichteter Haken ausgegeben. Zweitens sollten Sie darauf achten, bei Anwendung der hochauflösenden Grafik immer die Grafikfarbe mit zu spezifizieren. Diese wird durch das MODE-Kommando nicht verändert. Wenn Sie also in dem PLOT-Befehl in Zeile 10 die 1 als dritten Parameter weglassen und das Programm zweimal hintereinander mit RUN starten, so wird das Raumschiff beim zweiten Mal nicht mehr dargestellt. Grund: Am Anfang von Zeile 10 wird Farbe 2 auf 0 gesetzt. Vom ersten Durchgang her war als Grafikfarbe aber Register 2 spezifiziert.

Ergebnis: Das Raumschiff wird in diesem Durchgang mit der Hintergrundfarbe geschrieben. Nach der Zeitverzögerung wird zwar Farbe 2 aufgerufen, aber da Sie nun das ganze Raumschiff mit dieser Farbe geschrieben haben, blinkt auch die Rakete als Ganzes. Experimentieren Sie auch mit anderen Zeichen und Kombinationen.

(Carsten Straush/Matthias Rosin)



Mehr Komfort mit »Explora 2.0«

Der neue Checksummer für die Schneider-Computer ist da. Explora 2.0 macht die Eingabe von Programmen noch einfacher. Aber keine Angst, Explora 1.0 dürfen Sie auch noch benutzen.

Viele unserer Leser haben Explora 1.0 schon schätzen gelernt. Ist doch die Eingabe von Programmen bedeutend einfacher und sicherer geworden. Doch die erste Version unseres Checksummers zeigte noch einige Schwächen. Der Neue hat sie nicht mehr. Aber kompatibel zueinander, das sind sie beide.

Zuerst einmal Informationen für alle, die noch nicht wissen, was Explora ist. Explora ist ein Prüfsummenprogramm, das jede eingegebene Zeile auf ihre Richtigkeit hin überprüft. Dazu müssen Sie zuerst Explora laden (oder eintippen), einmal laufen lassen und dann wieder (mit »NEW«) löschen. Die Maschinencode-Routine im Speicher überprüft jetzt automatisch Ihre Eingaben. Sobald Sie am Ende einer Programmzeile ENTER oder RETURN drücken, erscheint in eckigen Klammern eine vierstellige Hexadezimalzahl auf dem Bildschirm. Das im Heft abgedruckte Listing enthält ebenfalls solche Zahlen. Stimmt die Prüfsumme auf dem Bildschirm mit der im Heft überein, so haben Sie die Zeile richtig abgetippt. Gibt es Unterschiede zwischen den Werten, müssen Sie auf Fehlersuche gehen und die Zeile noch einmal kontrollieren.

Explora 1.0 und 2.0 sind Brüder

Das alles konnte Explora 1.0 auch schon. Der Vorteil der neuen Version besteht darin, daß Sie nun größere Freiheiten bei der Eingabe der Zeilen haben. So dürfen Sie die Basic-Schlüsselwörter in Klein- oder Großbuchstaben (oder auch gemischt) eingeben. »PRINT« darf mit einem Fragezeichen abgekürzt werden. Explora 2.0 akzeptiert zum Beispiel für die Zeile »100 PRINT« folgende verschiedene Eingaben:

```
100 PRINT
100 print
100 ?
100 PrInT
```

Die Zeilen müssen also nicht mehr schon beim Eintippen wie abgedruckt aussehen, sondern erst beim Auflisten. Außerdem werden Prüfsummen nur noch für Programmzeilen ausgegeben und nicht mehr wie früher auch bei Direkteingaben. Vor der Zeilennummer stehende Leerzeichen, Line-Feeds und Tabulatorzeichen überliest Explora 2.0 selbständig. Leerzeichen innerhalb der Zeile zählen aber weiterhin und verändern die Prüfsumme.

Explora erlaubt die Verwendung des EDIT-Befehls. AUTO ist in der neuen Version ohne Einschränkungen zu



benutzen, allerdings nur beim CPC 664 und CPC 6128. Explora 1.0 lag im Speicher fest zwischen A000 und A086 hex. Die neue Version wird vom Basic-Lader automatisch im Speicher direkt unter HIMEM abgelegt. Somit arbeitet auch SYMBOL AFTER einwandfrei.

Allerdings sollten Sie Programme nur starten, wenn Sie vollständig eingegeben sind. Denn dann können Sie Explora aus dem Speicher eliminieren und es gibt auch bei Maschinencode-Routinen keine Platzprobleme.

Eine kleine Einschränkung gibt es aber doch: Löschen Sie keinesfalls Zeilen durch Eingabe der Zeilennummer und Drücken der ENTER-Taste. Die Zeile wird dann gar nicht wirklich gelöscht, sondern ist ein Duplikat der folgenden Zeile. Verwenden Sie statt dessen DELETE. Statt »20« müssen Sie also »DELETE 20« schreiben.

Das Wichtigste dürfen wir aber nicht vergessen! Explora 2.0 ist aufwärtskompatibel zu Explora 1.0. Das heißt, daß Sie sowohl mit Explora 2.0 alte Programme abtippen können als auch mit Explora 1.0 alle zukünftigen Programme. Denn die Prüfsummen sind identisch.

Die Prüfsummen sind identisch

Steuerzeichen und mehrere Leerzeichen, die in Strings aufeinanderfolgen, sind in geschweiften Klammern im Klartext angegeben. So bedeutet {CTRL A}, daß Sie die CTRL-Taste gemeinsam mit »A« drücken müssen. Aber Vorsicht, daß Sie solch ein Zeichen nicht mit den ASCII-Sonderzeichen { oder } verwechseln!

Die Bedeutung der geschweiften Klammer erkennen Sie jedoch sehr leicht. Denn als ASCII-Sonderzeichen steht sie meist allein. Im anderen Fall umschließt sie immer Control- oder Leerzeichen. Denn auch die Leerzeichen werden mit {5 SPACE} (fünf Leerzeichen) angezeigt.

Da die Tastatur der Schneider-Computer sehr leicht umdefiniert werden kann, arbeitet jeder Computer-Besitzer mit einer anderen Tastenbelegung. Wir verwenden nur den Original-ASCII-Code. Deutsche Sonderzeichen werden dabei als amerikanische Sonderzeichen (siehe Tabelle) dargestellt. Benutzen Sie einen umdefinierten Zeichensatz, so müssen Sie natürlich immer die analogen Zeichen einsetzen. Übrigens, das Zeichen »~« für das deutsche »ß« wird mit CTRL-2 aufgerufen.

Listing 1 enthält das Programm »Explora 2.0«. Listing 2 zeigt die Besonderheiten noch einmal mit Erklärung.

(Martin Kotulla/hg)

Programm-Steckbrief	
Name:	Explora 2.0
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

Sonderzeichen

amerikanische	deutsche
@	§
[	Ä
\	Ö
]	Ü
{	ä
|	ö
}	ü
~	ß

Das Symbol »ˆ« steht für »↑«

Tabelle der Sonderzeichen

```
100 ' **************************          [DFCC]
110 ' *                        *          [FADA]
120 ' *       EXPLORA V2.0     *          [761E]
130 ' *                        *          [DCDE]
140 ' **************************          [C3D4]
150 '                                     [E1BA]
160 DEF FNlsb(x)=255 AND UNT(x)           [39E0]
170 DEF FNmsb(x)=255 AND INT(x/256)       [8864]
180 SYMBOL AFTER 256:MEMORY HIMEM-161     [948C]
190 start=HIMEM+1:SYMBOL AFTER 240        [2092]
200 FOR i=&A000 TO &A09D:READ a$:sum=sum
    +VAL("&"+a$):NEXT i                   [B2C8]
210 IF sum<>19814 THEN PRINT "DATA-Fehle
    r!":END                               [FCCE]
220 RESTORE:FOR i=start TO start+&9D:REA
    D a$                                  [608E]
230 POKE i,VAL("&"+a$):NEXT i             [24D2]
240 FOR i=1 TO 5:READ a:a=a+start         [AC2A]
250 wert=PEEK(a)+PEEK(a+1)*256-40960+sta
    rt                                    [2776]
260 POKE a,FNlsb(wert):POKE a+1,FNmsb(we
    rt):NEXT i                            [01B2]
270 IF PEEK(6)=&80 THEN ed=&BD3A:POKE &B
    F20,&A4                               [56A8]
280 IF PEEK(6)=&7B THEN ed=&BD5B:POKE &B
    F20,&8A:RESTORE 470                   [760C]
```

Listing 1. Mehr Komfort mit »Explora 2.0«

```
290 IF PEEK(6)=&91 THEN ed=&BD5E:POKE &B
    F20,&8A:RESTORE 490                   [16FA]
300 POKE &BF21,&AC:POKE &BF22,PEEK(ed)    [71DE]
310 POKE &BF23,PEEK(ed+1):POKE &BF24,PEE
    K(ed+2)                               [9984]
320 POKE ed,&C3:POKE ed+1,FNlsb(start):P
    OKE ed+2,FNmsb(start)                 [9AE6]
330 IF PEEK(6)=&80 THEN END               [6044]
340 FOR i=1 TO 7:READ a$,b$:a=VAL("&"+a$
    )+start:b=VAL("&"+b$)                 [3306]
350 POKE a,FNlsb(b):POKE a+1,FNmsb(b):NE
    XT i                                  [0332]
360 DATA CD,22,BF,F5,C5,D5,E5,2A,20,BF,C
    D,61,DD,B7,28,62                      [5BFC]
370 DATA E5,2A,20,BF,CD,88,A0,E1,30,58,C
    D,04,EE,CD,A3,E7                      [5EF2]
380 DATA CD,63,E1,ED,4B,20,BF,21,00,00,0
    A,5F,16,00,19,03                      [DBF6]
390 DATA FE,00,20,F6,DD,2A,20,BF,01,00,0
    0,DD,7E,00,5F,16                      [4D3E]
400 DATA 00,19,04,F5,A8,47,F1,09,DD,23,F
    E,00,20,ED,3E,0D                      [E53C]
410 DATA CD,5A,BB,3E,0A,CD,5A,BB,3E,5B,C
    D,5A,BB,7C,CD,77                      [259A]
420 DATA A0,7C,CD,7B,A0,7D,CD,77,A0,7D,C
    D,7B,A0,3E,5D,CD                      [014A]
430 DATA 5A,BB,E1,D1,C1,F1,C9,1F,1F,1F,1
    F,E6,0F,C6,30,FE                      [A10A]
440 DATA 3A,38,02,C6,07,C3,5A,BB,CD,61,D
    D,B7,37,C8,CD,04                      [64AC]
450 DATA EE,D0,7E,FE,20,20,01,23,CD,D2,E
    6,37,9F,C9                            [0C36]
460 DATA &15,&5F,&63,&67,&6B              [3A22]
470 DATA 0B,DE52,1B,EED4,1E,E869          [7B14]
480 DATA 21,E259,89,DE52,8F,EED4,99,E7AA
                                          [0586]
490 DATA 0B,DE4D,1B,EECF,1E,E864          [1F52]
500 DATA 21,E254,89,DE4D,8F,EECF,99,E7A5
                                          [249A]
510 END                                   [AA1A]
```

Listing 1. Mehr Komfort mit »Explora 2.0« (Schluß)

```
400 DATA &BB,&E1,&D1,&C1,&F1,&C9,&1F,&1F
    ,&1F,&1F,&E6,&0F,&C6,&30,&FE,&3A      [C4C0]
410 DATA &38,&02,&C6,&07,&C3,&5A,&BB      [633C]
420 NEW                                   [EE40]
430 PRINT "{CTRL A}{CTRL Y}{CTRL Y}{CTRL
     A}"                                  [8DB8]
440 PRINT "{5 SPACE}WW}{"                 [8CDE]
```

Listing 2. Im Beispiel müssen Sie die Zeile 400 wie folgt eingeben (Modus 1): 400 DATA &BB,$E1,&D1,&C1,&F1,&C9, &1F,&1F ,&1F,&1F,&E6,&0F,&C6,&30,&FE,&3A Zeile 430 besteht in der PRINT-Anweisung aus den vier Tastendrucken CTRL-A, CTRL-Y, CTRL-Y und CTRL-A. Der String in Zeile 440 lautet »WW}{«

Interpreter-»Fehler« – selbst behoben

Das Betriebssystem und speziell das Basic der Schneider-Computer gehört zum Besten, was es in dieser Art zur Zeit auf dem Heimcomputermarkt gibt. Allerdings haben auch die CPCs noch einige Macken. Hilfe ist oft sehr einfach.

Daß bei der Entwicklung der Software der Schneider-Computer gute Arbeit geleistet wurde, ist allgemein bekannt. Aber auch das beste Betriebssystem weist immer noch einige Fehler auf. So auch das Schneider-Basic. Und dies sind nicht nur schon relativ bekannte »Macken«, wie der DEC$-Befehl, sondern oft Kleinigkeiten, die anfangs zu einiger Verwunderung führen und sich erst nach genauerem Hinsehen als echte Fehler – und nicht etwa Programmierfehler – entpuppen. Leicht zu verstehen ist der gerade beschriebene Fehler beim DEC$ des CPC 464. Dieses Kommando dient der Formatierung eines Strings. Das heißt, es baut eine vorgegebene Zeichenkette so um, als ob diese mit einem PRINT-USING-Kommando ausgegeben würde. Leider unterlief bei der Entwicklung des Betriebssystems beim 464 ein Fehler.

Zwei auf – eine zu

Die Anfangsklammer wird zweimal überprüft, weshalb man bei diesem Kommando immer zwei Klammern öffnen muß, aber nur eine schließen darf. Die Funktion DEC$ hat dabei folgendes Format:

```
a$=DEC$((a$,"<Formatierungs-
parameter>")
```

Die Formatierungsparameter entsprechen denen des PRINT-USING-Befehls. Nähere Angaben zur Verwendung dieser Parameter finden Sie im Handbuch.

Während die Falsch-Interpretation bei DEC$ eindeutig erkennbar ist, bereiten andere Fehler mehr Schwierigkeiten. Ein Problem im mathematischen Bereich stellt beispielsweise die Potenzfunktion dar. Wenn man hier mit einer negativen Basis arbeitet, liefert der CPC 464 in manchen Fällen ein falsches Ergebnis. Laut Handbuch ist die Potenzfunktion b^e in zwei Fällen definiert.

1. Wenn die Basis (b) positiv ist, soll die Potenzfunktion für beliebige Exponenten (e) definiert sein.
2. Bei negativer Basis ist sie nur für ganze Exponenten definiert.

Der Exponent darf dabei sowohl positiv als auch negativ sein. Probleme hat der Interpreter dabei mit einem an sich völlig einfachen Problem. Was passiert, wenn man eine negative Zahl mit 1 potenziert? Sie wird positiv – jedenfalls beim 464. Bei den Nachfolgemodellen wurde dies kuriert. Wenn Sie also

```
PRINT(-12)↑1
```

eingeben, so lautet das Ergebnis nicht -12, sondern 12. Der Schneider interpretiert hier also die negative Basis als positiv und rechnet mit dem positiven Ansatz weiter. Eine einfache Abhilfe bietet die Multiplikation mit der Signum-Funktion. SGN(<Wert>) liefert –1 für einen negativen Wert, 0 bei Null und +1 für einen positiven Eingabewert. Wenn Sie also beim 464

```
PRINT SGN(b)*b↑e
```

statt des normalen Funktionsaufrufes benutzen, erhalten Sie immer die richtige, korrigierte Ausgabe. Hierzu allerdings gleich eine Nebenbemerkung. Sie dürfen den gerade beschriebenen Fehler des Computers – genauer seines Interpreters – nicht mit einem anderen »Fehler« verwechseln, der auf den ersten Blick auch zu Verwunderung führt. Wenn Sie nämlich

```
?-12↑2
```

eingeben, erhalten Sie – und dies gilt nun gemeinsam für alle Schneider-Computer – nicht etwa, wie Sie vielleicht erwarten, 144, sondern vielmehr -144, also den erwarteten Wert mit negativem Vorzeichen. Dies ist aber kein Interpreterfehler, sondern liegt in der Hierarchie der einzelnen Verknüpfungssymbole begründet. Sie kennen sicher die mathematische Grundregel »Punkt-vor-Strich-Rechnung«. Diese richtig anwendend, würden Sie beispielsweise den Term »2*3–3*3« als »(2*3) – (3*3)« berechnen. Also zunächst die mit dem Malzeichen verbundenen Werte bestimmen und diese dann mit »–« verknüpfen. Analog dazu werden Potenz- und andere Funktionen vor der Verknüpfung mit »*« oder »/« berechnet. Auch der Schneider macht das so. Er gliedert den eingegebenen

Hierarchie der Rechenzeichen

Ausdruck mittels der einzelnen Verknüpfungszeichen (»+«, »–«, »*«, »/« und Funktionsaufrufe). Dabei gilt folgende Hierarchie:

1. Klammern: haben höchste Priorität
2. Funktionsaufrufe: SIN(), SQR() und so weiter werden zuerst bestimmt
3. Potenzfunktion »↑«
4. »–« negatives Vorzeichen für eine Zahl
5. »*« Multiplizieren
6. MOD Modulo-Funktion (liefert den Rest einer Division)
7. »/« Dividieren
8. »\« Ganzzahldivision (nur der ganzzahlige Wert des Ergebnisses wird angezeigt)
9. »+« Addition
10. »–« Subtraktion

Das Rechensymbol mit der niedrigsten Zahl hat dabei die höchste Priorität. Wie Sie sehen, kann dabei das Minuszeichen zwei verschiedene Bedeutungen einnehmen. Zum einen kann es die Operation »–«, also eine Subtraktion, anzeigen. Ferner kann das Minuszeichen als Vorzeichen stehen. Diese typisch menschliche Doppeldeutigkeit macht dem Schneider nun aber arg zu schaffen. Er hat das Minus in beiden Versionen gespeichert, erkennt es auch richtig, und führt demnach dann die richtige Operation aus. Allerdings setzt er das Vorzeichen in seiner Prioritätsbestimmung geringer als Funktio-



nen an. Während ein Mensch also -12^2 – mathematisch falsch – als Aufforderung verstehen würde, die Zahl -12 mit sich selbst zu multiplizieren, denkt der Schneider hier korrekt. Er analysiert das Minuszeichen und bestimmt, daß zu seiner Definition eine Minusfunktion auf den Ausdruck, der nach diesem Zeichen steht, angewendet werden soll. Wenn Sie also

```
PRINT-12↑2
```

eingeben, so interpretiert der Schneider mathematisch korrekt: »Zahl 12 mit sich selbst multiplizieren und darauf dann die Minusfunktion anwenden – also das durch die Potenzierung entstandene Ergebnis mit -1 multiplizieren.« Das Endergebnis lautet also -144.

Überhaupt ist der Umgang mit Zahlen bei Computern – und hier machen die Schneider-Geräte keine Ausnahme – eine Wissenschaft für sich. Ein weiterer Punkt, der oft zu Ärger führt, ist die teilweise ungenügende Rechengenauigkeit – speziell bei kleinen Zahlen. Oft will man in einer Schleife mit relativ hoher Auflösung nahe beieinander liegende Punkte berechnen, um beispielsweise eine Funktion zu plotten. Dann darf man sich aber nicht mehr auf seinen Computer verlassen, weil dieser ungenauer arbeitet, als das Problem es erlaubt. Nehmen wir ein ganz einfaches Beispiel:

```
10 FOR i=0.1 TO 10 STEP 0.1
20 PRINT i
30 NEXT i
```

Was erwarten wir bei einer solchen Schleife? Das Programm beginnt mit i=0,1. Das ist der erste Schritt, bei dem als Beispiel der erste darzustellende Wert berechnet werden soll. Es folgt 0,2, 0,3 und so weiter. Bei 1,0 haben wir die ersten 10 Werte berechnet, bei 2,0 20 und demnach bei 10,0 genau 100. Nun schauen Sie sich einmal an, was Ihr CPC berechnet hat. Am Anfang ist alles noch korrekt. Auf 0,1 folgt 0,2.

Eins und eins ist nicht immer zwei

Ab 3,7 wird es jedoch ungenau. Jetzt gibt es als nächste Stufe 3.79999999. Das wäre nicht weiter schlimm, wenn der Schneider in seiner Vergleichsbetrachtung, mit der er feststellt, ob der letzte Schleifenwert bereits erreicht oder überschritten wurde, genauso wenig pingelig wäre und diese Werte entsprechend runden würde. Hier verhält er sich aber nun extrem kleinlich. 3.7999999 ist kleiner als 3.8 und obwohl es eigentlich 3.8 sein muß, fährt der Schneider noch einmal durch die Schleife. Die FOR-TO-Schleife wird ja bekanntlich abgebrochen, wenn durch laufende Additionen ein größerer Wert als der Schleifenendwert erreicht wird. Dies gibt keine Probleme, wenn die Differenz (der Abstand, in dem wir die einzelnen Werte ausdrucken) groß im Vergleich zu der Abweichung ist. Es ist nämlich egal, ob wir zu 3.99999 oder zu 4 den Wert 0.1 addieren. Ist 4 unser Schleifenendwert, fällt der Vergleich immer positiv aus. Der Computer beendet die Schleife. Gehen wir aber beispielsweise in Tausendstel-Schritten voran, so kann es durchaus passieren, daß der CPC ein paar Berechnungen mehr macht als eigentlich vorgesehen.

99 statt 100

Nun aber subtrahiert der Schneider nicht nur, manchmal fügt er auch ein bißchen dazu. Am Ende unserer Schleife (ab 9.7) wird das klar. Und wenn er gerade zu glatten Werten zurückgekehrt ist, legt er noch ein bißchen drauf. Der CPC entscheidet, daß 9.9000001 + 0.1 bereits größer als 10 und damit der Schleifenendwert 10 überschritten ist. Die Schleife muß verlassen werden. Leider ist dies ein Schritt zu früh. Statt 100 Durchgängen hat der Schneider nur 99 gemacht. Um diesem Problem abzuhelfen, gibt es zwei Methoden. Zum einen kann man den Teufel mit dem Beelzebub austreiben, genauer: die eine Ungenauigkeit durch eine geplante zweite ausgleichen. In unserem Fall heißt das, den Schleifenendwert auf 10,000001 zu erhöhen, womit auch noch der letzte Vergleich positiv ausfällt und die gewünschten 100 Schleifendurchgänge stattfinden. Dies ist aber eine wenig schöne Variante. Besser ist es, sich mit dem Grund für diese Ungenauigkeiten auseinanderzusetzen. Wie jeder Computer rechnet auch der Schneider intern binär – das heißt im Zweiersystem. Um nun mit uns Menschen zu kommunizieren, formt er seine Binärwerte in unsere Dezimalzahlen um. Es gibt aber Dezimalzahlen, die im Dezimalsystem ganz glatt aussehen, binär jedoch zu einem unendlich langen Bandwurm werden – wie beispielsweise 0,1. Binär lautet dieser Wert 0,000110110110110110110... Der Computer muß hier runden. Damit schleichen sich Ungenauigkeiten ein. Bei hundert aufeinanderfolgenden Additionen kommt Fehler zu Fehler, und am Schluß stimmt das Ergebnis nicht mehr. Hier gibt es aber eine relativ einfache Lösung. Statt »i« in solch binär schwer faßbaren Werten zu verändern, addieren Sie besser Ganzzahlen und setzen die Schleifenobergrenze entsprechend herauf. Die richtigen Werte von i für die Verarbeitung erhalten Sie dann durch einfaches Teilen – beispielsweise durch 10. Wenn Sie also statt der ersten Schleife mit

```
10 FOR i=1 TO 100
20 PRINT i/10
30 NEXT
```

arbeiten, so haben Sie exakt dieselbe Zahl an Schleifendurchläufen zur Verfügung. Nur die Additionsfehler werden vermieden. Denn 1 dezimal ist auch für den Computer 1 binär und damit problemlos zu verarbeiten.

Die Schneider machen jedoch nicht nur bei der Arithmetik manches anders, als man sich wünscht. Zwei weitere Probleme treten auf, wenn man sich intensiv mit dem Ablegen von Programmen beschäftigt. In einem vernünftigen Programm muß der Benutzer den Namen einer zu speichernden Datei eingeben können. Dazu reicht eine einfache INPUT-Abfrage. Wenn man mit

```
INPUT"Name der Datei";n$
```

einen Namen einliest, sollte man diesen problemlos mit OPENOUT n$ an die Disketten- oder Kassettenverwaltung weiterleiten können. Diese soll dann die Datei unter dem Namen speichern. Das klappt aber nur manchmal. Wenn man viele Strings im Speicher verwaltet, streikt der CPC. Statt eine Datei zu öffnen, wirft er eine Fehlermeldung aus.

Wirkung ohne Zeichen

Abhilfe ist auch hier wieder relativ einfach. Man muß nur vor dem String ein »""« ausgeben. Obwohl der Dateiname durch OPENOUT ""+n$ natürlich nicht geändert wird, akzeptiert der Computer nun das Kommando.

Ein weiterer Fehler wird Ihnen auffallen, wenn Sie des öfteren mit RENUM arbeiten, um sich Platz für neue Zeilen zu schaffen. RENUM renumeriert »fast alles«. Eine Zeilenangabe läßt der Befehl jedoch aus – die Zeilenangabe bei CHAIN MERGE. Mit diesem Kommando ist es möglich, ein Programm zu einem aktuellen Programm hinzuzuladen. Das neue Programm fügt sich dabei nahtlos in das alte ein. Zeilen mit gleicher Zeilennummer werden überschrieben. Durch eine nachfolgende Zeilenangabe kann man dabei den CPC anweisen, nach dem Ladevorgang das neuentstandene Programm ab einer beliebigen Stelle aufzurufen. Das Format lautet dabei:

```
CHAIN MERGE "Dateiname",<Zeile>
```

Diese nachgestellte Zeilenangabe verändert RENUM nicht. Wenn Sie also ein Gesamtprogramm mit CHAIN MERGE aus mehreren Teilen zusammensetzen wollen, so müssen Sie die Startadresse gegebenenfalls von Hand anpassen. (Carsten Straush/hg)

Basic – Stück für Stück

Jedes Programm in diesem Sonderheft enthält eine Programmbeschreibung. Für den Einsteiger bleiben aber trotzdem oft noch Fragen offen. Deshalb erklären wir Ihnen hier ein Listing Befehl für Befehl.

Nun soll aber nicht nur das Programmieren auf den Schneider-Computern gezeigt werden, sondern am Ende wollen wir auch über ein lauffähiges, praktisches Programm verfügen. Das Listing »Band-Kartei« ist ein Hilfsprogramm, das Ordnung in den Band- beziehungsweise Diskettensalat bringt. Wenn Sie nur einen 464 ohne Diskettenstation besitzen, kennen Sie das Problem sicherlich. Aber auch die Eigentümer der Floppy-bestückten großen Brüder werden es im Laufe der Zeit – mit zunehmendem Diskettenbestand – kennenlernen. Der Anlaß zu diesem Programm ist ein bemerkenswertes Computergesetz: Eine Datei ist immer gerade da, wo man sie nicht vermutet. Also sucht man geduldig auf seinen Bändern und Disketten hin und her, und manchmal findet man das Gesuchte dann auch. Man kann sich hier natürlich mit beigelegten Zettelchen oder Aufklebern etc. helfen. Aber diese sind oft nicht mehr »up to date«, weil die Diskette schon dreimal wiederverwendet worden ist. Beipackzettel haben dagegen die Eigenschaft, permanent zu verschwinden. Was fehlt, ist eine Art Katalogisierungsprogramm, das alle Bänder und Disketten enthält. Wenn man ein bestimmtes Programm sucht, muß man nur noch seinen Namen eingeben und schon erhält man die Bandnummer und den Zählwerkstand, wo sich dieser Titel befindet, auf den Bildschirm. Zu schön um wahr zu sein? Nein, wir müssen nur ein wenig Programmierarbeit leisten, um dieses Ziel zu erreichen.

Machen wir uns zunächst ein paar Gedanken darüber, was ein solches Programm eigentlich alles ausführen soll. Vor allem benötigen wir ein umfangreiches Inhaltsverzeichnis. Für jedes Band, beziehungsweise jede Diskette, sollen dabei die einzelnen gespeicherten Dateien abrufbar sein. Nun können wir aus Platzgründen nicht den Inhalt aller abgespeicherten Bänder gleichzeitig auf den Bildschirm bringen. Wir beschränken uns also am besten auf den Inhalt eines Bandes. Um etwas Ordnung in unsere Dateien zu bekommen, überlegen wir uns die benötigten Daten und deren maximale Größe und Anzahl. Das beste Programm nützt nämlich wenig, wenn am Schluß Probleme mit dem Speicherplatz auftreten. Außerdem kann der Schneider keine unendlich großen Datenfelder verwalten. Er braucht eine Maximalangabe über deren Größe, die sogenannte Dimensionierung.

Die erste Größe kennen wir bereits schon, die maximale Zahl der Bänder, die erfaßt werden soll. Die nächste Dimensionierung betrifft die einzelnen Programmtitel. Hier gehen wir von 20 zu katalogisierenden Bändern mit je maximal 15 Titeln aus. Für deren Ablage bietet sich bei der Basic-Programmierung ein zweidimensionales Datenfeld, ein sogenanntes Array an. Bezeichnen wir es mit n$. Die erste Dimension, also der erste Index, soll dabei die Nummer des Bandes sein. Index Nummer 2 ist die Nummer des Programms auf dem Datenträger. n$(1,3) bezeichnet damit das dritte Programm auf Band Nummer 1, n$(3,4) bezeichnet die vierte Datei auf Band Nummer 3. Die einzelnen Elemente unseres Datenfeldes sind einfache Strings, die die verschiedenen Programmtitel enthalten sollen.

Felder der Dimension 300 haben 336 Elemente

Nun wissen Sie sicher, daß, wenn wir ein Feld auf 20*15 dimensionieren, dieses aus 21*16 Elementen besteht, denn der Schneider rechnet ab Null. n$(Bandnummer,1) bis n$(Bandnummer,15) enthalten die Programmtitel. n$(Bandnummer,0) bleibt damit aber noch frei. Da wir aber beabsichtigen, noch einen weiteren String je Band abzuspeichern – irgendwo soll ja schließlich auch noch der Name des Bandes zu finden sein – kommt uns dies ganz gelegen. Der Bandname wird jeweils in n$(Bandnummer,0) abgelegt.

Nun fehlt noch der Zählwerkstand, der auch wichtig ist, um einen gezielten Zugriff auf ein Band zu ermöglichen. Dazu benötigen wir noch ein ebenso großes numerisches Array, das für jedes Programm den Zählwerkstand aufnimmt. Es trägt die Bezeichnung n. Wieder bleibt der Nullindex unbelegt. Und auch hier erhält er einen neuen Sinn: Er beinhaltet die Anzahl der auf einem Band befindlichen Programme.

Der Datenrahmen ist nun festgelegt. Bevor wir uns jedoch nun direkt auf das Listing stürzen, kurz noch ein paar Bemerkungen zu den einzelnen Funktionen. Das Programm »Band-Kartei« stellt im wesentlichen eine Art komfortables Inhaltsverzeichnis dar. Dabei erscheint jeweils der Inhalt eines Bandes auf dem Schirm. Da wir aber natürlich viele Bänder abspeichern wollen – sonst hat das Ganze ja gar keinen Sinn! – müssen wir einige Funktionen einbauen, die uns das Umschalten zwischen den einzelnen Bändern ermöglichen. Dabei müssen wir sowohl in den Datensätzen blättern, also auch die nachfolgenden Programm-Inhaltsverzeichnisse ausgeben lassen, als auch gezielt nach einem Programm oder einem Band suchen. Außerdem brauchen wir natürlich Prozeduren, die uns das Löschen, Einfügen oder Ändern eines Programmtitels oder Bandes gestatten.

Das Programm ist nach einem Menüsystem aufgebaut. Die einzelnen Programmfunktionen bilden dabei eine Art Kreisverkehr. Wenn wir beispielsweise eine Datei laden wollen, so geschieht dies mit der Funktion Nummer 1 »Datei einlesen«. Aufgerufen wird sie aus dem Auswahlmenü ab Zeile 250. Das Programm verzweigt jeweils mit GOSUB in die entsprechende Unterroutine. Der Menüunterpunkt »Verlassen« führt in die jeweils übergeordnete Ebene zurück. Der Punkt »Verlassen« im Hauptmenü veranlaßt einen Programmabbruch.

Gehen wir nun einmal das Programm Zeile für Zeile durch. Die einzelnen Unterpunkte sind dabei anhand der kommentierenden REM-Anweisungen gut erkenntlich. Den Anfang bildet ein Initialisierungsteil. Hier werden die Farben festgelegt. Die Farbregister 0 bis 3 enthalten die gewünschten Werte (schwarz, rot, gelb und mittelblau). In Zeile 50 erfolgt das Umschalten auf den Modus 0 (dies sorgt für eine bessere Wirkung der nachfolgenden Titelausgabe) und die Ladegeschwindigkeit 1 wird festgelegt. Die folgenden Zeilen geben den Titel aus. Die Sternchen bewirkt dabei das PEN-Kommando mit INK 3, also rot. Interessant ist hier noch in Zeile 100 die Farbumschaltung innerhalb einer Bildschirmzeile. Das »;« nach dem ersten PRINT-Kommando sorgt dafür, daß der Cursor an der letzten Position stehen bleibt. Dann wird mit PEN 1 gelb gewählt, der Titel ausgegeben und dann nach demselben Prinzip wieder zurückgeschaltet. Die vielen PRINT-Kommandos in den Zeilen 60 und 140 bewirken dabei nichts anderes als einen einfachen Zeilenvorschub. Der Cursor bewegt sich also auf die nächste Zeile. Die Zeilen 150-190 stellen eine einfache Warteschleife dar.

Angewandt wurde hier der Befehl WHILE/WEND. Solange die nach dem WHILE stehende Bedingung erfüllt ist, arbeitet der Schneider bei jedem WEND in der nach dem WHILE-Kommando stehende Zeile weiter. Als Bedingung ist i<2000 vorgegeben, das heißt bei jedem Schleifendurchlauf wird i jeweils um 1 erhöht, bis 2000 erreicht ist (Zeile 180). Spätestens nach 2000 Durchgängen erfolgt also kein Rücksprung mehr und es geht in Zeile 200 weiter. Es gibt allerdings auch noch eine andere Abbruchbedingung. Wenn Sie nämlich eine beliebige Taste drücken, wird durch den INKEY-Befehl in Zeile 160 z$ mit dem Tastenwert belegt und es erfolgt direkt aus Zeile 170 ein Aufruf von Zeile 200. Dann dauert es nicht mehr so lange, bis es im Programmtext weitergeht. Die Zeitverzögerung hat übrigens den Sinn, Ihnen Zeit zu geben, den Bildschirminhalt gründlich zu studieren. Im nachfolgenden Teil wird nun durch MODE 1 und die Definition von drei Bildschirmfenstern der Bildschirm gelöscht und völlig umgebaut. Ohne diese Zeitverzögerung wäre vom Titel nur ein kurzes Aufblitzen zu sehen.

Top-Window ist top

Der Rest unseres Programms arbeitet mit der neuen Bildschirmgestaltung. Drei Zeilen als Top-Window dienen dazu, Überschriften aufzunehmen. Im unteren Teil des Schirms wurden analog dazu drei Zeilen für die Eingabe reserviert (WINDOW#2). Der Hauptbildschirm (WINDOW#0) ist damit auf die mittleren 19 Zeilen beschränkt. Nachdem den einzelnen Bildschirmfenstern in Zeile 230 und 240 die Werte für Schrift- und Hintergrundfarbe zugeordnet wurden, geht es ins Auswahlmenü. Zunächst werden durch die CLS-Befehle die drei Bildschirmfenster gelöscht. Danach folgt die Ausgabe des Titels in der Mitte von Fenster 1. Anschließend erscheinen in der Mitte des Hauptbildschirms die einzelnen Routinen. Zeile 350 prüft, welche Wahl der Benutzer getroffen hat. Die Tastaturabfrage erfolgt mit INKEY$ und es wird festgestellt, ob überhaupt eine Taste gedrückt wurde. Ist dies nicht der Fall (Z$=""), geht es direkt zurück an den Zeilenanfang. Dieses Spielchen wiederholt sich so lange, bis eine Taste betätigt wird. Dann überprüft ASC, ob eine gültige Zahl eingegeben wurde. Die vorherige Abfrage auf "" ist dabei unbedingt notwendig. Die ASC-Funktion verkraftet nämlich keinen Leerstring. PRINT ASC("") hat einen IMPROPER ARGUMENT-Fehler zur Folge. Nachdem die Eingabe auf Stimmigkeit überprüft wurde, wird in Abhängigkeit vom eingegebenen Wert zu einem der verschiedenen Unterprogramme verzweigt. Zeile 360 mit den verschiedenen GOSUB-Anweisungen bewirkt in unserem Lernprogramm die entsprechenden Verzweigungen.

Das erste Programm, auf das wir hierbei treffen, ist die Datei-Leseroutine. Sie beginnt bei Zeile 410. Ihr Ablauf ist schnell erklärt. Nach Ausgabe des Funktionstitels in Zeile 410 wird eine Datei zum Lesen geöffnet und dann als erstes die Variable md eingelesen. md steht für »maximale daten« und entspricht der Anzahl der abgespeicherten Bänder. Es wäre ja unsinnig, bei beispielsweise fünf erfaßten Datenträgern die Maximalzahl von 20 Dateien wegzuschreiben. In einer doppelten Schleife werden dann alle benötigten Daten eingelesen. i läuft dabei von 0 bis md, also über alle Bänder. Zunächst werden dabei in Zeile 440 der Name des Bandes und die Anzahl der abgespeicherten Titel auf diesem Band abgefragt. Danach folgt die j-Schleife, wobei j von 1 bis n(i,0), also wieder über alle Titel, läuft. Damit werden für jedes Band (i) alle Titel (j) eingelesen. Zwei NEXT-Befehle schließen diese Schleife ab. Die Lesedatei beendet CLOSEIN (dies ist unbedingt notwendig, damit man beim nächsten Laden keine Probleme bekommt), und es geht zurück ins Hauptmenü. Wenn Ihnen in diesem Programmteil noch irgend etwas unklar ist, so erhöhen Sie einmal die Schleifenwerte für i und j nacheinander und schauen sich an, was das zur Folge hat.

Die nächste Routine benötigt keinerlei Erklärungen, da sie ein Pendant der ersten darstellt. Sie haben nur in der oben beschriebenen Leseroutine alle INPUTs durch PRINT zu ersetzen und schon ist auch der Schreibteil fertig. Die Funktionsweise verläuft damit analog der ersten, nur in der umgekehrten Richtung.

Nun kommen wir zu einem etwas größeren Programmteil – der Druckausgabe. Diese hält drei Routinen bereit. Man kann alle verfügbaren Bänder oder Programme ausdrucken, beispielsweise um eine Übersicht zu erhalten oder ein spezielles Band herauszugreifen. Den dazu benutzten Auswahlmechanismus kennen Sie bereits. Nach der Ausgabe des Funktionskopfes wird in einer INKEY$-Schleife (Zeile 680) die gewünschte Funktion festgestellt. »3« führt zurück ins Hauptmenü, Werte ungleich »1« und »2« zurück an den Zeilenanfang. Nun fällt die Entscheidung zwischen der Gesamtausgabe und dem Einzelbandausdruck an. Ab Zeile 700 sehen Sie die Funktionsweise. Zuerst kommt die Variante Gesamtausdruck an die Reihe. Es wird die Aufforderung zum Einschalten des Druckers ausgegeben und dann mit einer WHILE/WEND-Schleife die Bestätigung abgewartet. Diese einfachste Kontroll-Abfrage mit WHILE/WEND ist sehr kurz und komfortabel.

Drucken mit Doppelschleife

Es folgt der eigentliche Druckteil, der Sie vielleicht ein wenig an den Teil zur Datenspeicherung beziehungsweise zum -lesen erinnert. Das Ausgabeprinzip geht weitgehend identisch vor sich. Mit Hilfe der schon bekannten i,j-Doppelschleife werden zuerst (in Zeile 730) der Bandtitel und die Anzahl der abgespeicherten Programme ausgegeben, dann mittels j die auf diesem Band abgespeicherten Programme. Vielleicht wundern Sie sich bei dieser Prozedur, warum alle Ausgaben mit PRINT#d getätigt werden. Dies hat einen einfachen Grund. d wird in Zeile 680 auf 8 gesetzt und lenkt damit den Ausdruck normalerweise auf das Gerät Nummer 8, den Drucker. Wenn Sie aber in dieser Zeile d=0 setzen, wird die Ausgabe auf den Hauptbildschirm umgeleitet. Sie brauchen beim Experimentieren und bei den ersten Laufversuchen mit diesem Programm dann nicht immer den Drucker einzuschalten und meterweise Papier bedrucken. Zeile 780 führt zurück ins Hauptmenü und beendet den Teil Gesamtausdruck.

Die folgenden Zeilen dienen dem gezielten Einzelausdruck. Zuerst muß der Computer wissen, was er überhaupt drucken soll. Dazu wird zunächst in Zeile 790 der Name des betreffenden Bandes, dessen Inhalt wir erfahren wollen, abgefragt. Es folgt eine kleine Suchroutine. Dabei wird Band für Band (über alle i von 0 bis md) überprüft, ob der eingegebene Name mit dem gespeicherten Bandtitel übereinstimmt. Da der CPC zwischen Groß- und Kleinschreibung unterscheidet, sind dabei sowohl der Suchbegriff als auch die zu vergleichenden Bandtitel in Kleinschrift zu konvertieren. Dazu dienen die LOWER$-Befehle in Zeile 790 und 810, wobei in der letztgenannten Zeile auch der eigentliche Vergleich stattfindet. Solange dieser negativ ausfällt, wird der nächste Bandtitel untersucht. Es gibt damit nun zwei Möglichkeiten, um nach Zeile 820 zu gelangen. Möglichkeit 1, es hat keine Übereinstimmung stattgefunden. In diesem Fall wird die FOR-NEXT-Schleife bis zum Ende abgearbeitet. Im zweiten Fall wurde Gleichheit festgestellt. Jetzt war die Ungleichheitsbedingung in Zeile 810 nicht mehr erfüllt und deshalb erfolgte auch kein NEXT mehr. Die

Schleife wurde also nicht vollständig abgearbeitet.

Wir können nun den Zustand der Schleifenvariablen i benutzen, um zwischen den beiden Möglichkeiten zu unterscheiden. Nach Ausführung einer FOR-TO-Schleife weist die Schleifenvariable – in unserem Fall also i – einen um den STEP-Wert höheren Inhalt als die Schleifengrenze auf. Da wir kein STEP spezifiziert haben, i sich also in Schritten zu 1 erhöht, hat i in diesem Fall den Wert md+1, ist also größer als md. Wir können daher mit einer Abfrage wie in Zeile 820 feststellen, ob die Schleife bereits bis zum letzten NEXT bearbeitet wurde. Dann wird eine Negativmeldung ausgegeben und nach einer Zeitverzögerung ins Hauptmenü zurückgekehrt. Ansonsten enthält i die Nummer des gesuchten Bandes in der Datei. Die Ausgabe des Bandtitels und der einzelnen Programmnamen geschehen dann wie bei der Gesamtdruckvariante mit Hilfe einer einfachen Schleife. Von besonderer Bedeutung ist in diesem Zusammenhang nur noch die Formatierung des Ausgabestrings in den Zeilen 760 und 870. Es sollen die Programmnummer, der Programmname und der zugehörige Zählerstand ausgegeben werden. Dazu benutzen wir die Funktion DEC$, die einen formatierten String aus einer Zahl erzeugt und die LEFT$-Funktion, die die Länge des auszugebenden Titels auf 22 Zeichen beschränkt. Hierzu noch eine Besonderheit: Der DEC$-Befehl wird aufgrund eines Betriebssystemfehlers beim CPC 464 nicht richtig bearbeitet. Man muß daher bei diesem Computertyp zwei Klammern öffnen, damit diese Funktion läuft. So ist DEC$ dann auch im Listing abgedruckt. Wenn Sie das Programm also auf einem 664 oder 6128 laufen lassen, müssen Sie die zweite Öffnungsklammer streichen. Bei diesen Computern wurde der Betriebssystem-Fehler behoben.

Mit der Druckroutine haben wir jetzt die Nebenprogramme abgeschlossen und kommen zum eigentlichen Kern unseres Programms, den Datenbearbeitungsroutinen. Aus dem Hauptmenü werden all diese Funktionen mit der Nummer 3 aufgerufen. Da dies eine ganze Reihe ist, machte ein ganzes neues Menü mit Unterfunktionen erforderlich. Sie finden es ab der Zeile 630 im Listing. Alle Unterroutinen, mit denen wir uns nun beschäftigen, kehren zuerst einmal in dieses Menü zurück. Nur die Funktion 4 (Verlassen) führt zum eigentlichen Hauptmenü.

Ab Zeile 930 steht zunächst einmal die Ausgabe des Funktionstitels und des aktuellen Bandes auf dem Programm. Zeile 940 gibt dazu die Dateinummer und die Anzahl der in ihr enthaltenen Programmtitel aus, sowie den Dateinamen. Wichtig ist dabei die Variable ak. Sie enthält die Nummer des Bandes, mit dem momentan (aktuell) gearbeitet wird. Dementsprechend findet sich unter n$(ak,0) der Bandname, unter n(ak,0) die Anzahl der abgespeicherten Titel und so weiter. Nach der Titelausgabe wird das Inhaltsverzeichnis des aktuellen Bandes in einer Schleife ausgegeben (Zeile 960-990). Die Formatierung entspricht dabei der Druckroutine. Nachdem wir das gesamte Inhaltsverzeichnis auf dem Schirm haben, präsentiert Zeile 1000 das Menü. Die Abfragetechnik ist auch schon vom Hauptprogramm her bekannt: INKEY$-Schleife mit nachfolgendem »GOSUB-Sprungverteiler«. Zeile 1030 gibt nach der Rückkehr aus einem beliebigen Unterprogramm wieder an den Titelteil zurück. Das stellt sicher, daß, falls in einem Unterprogramm ein neues Band gewählt oder die alte Bildschirmausgabe zerstört wurde, diese vor jeder weiteren Abfrage erst einmal wieder restauriert beziehungsweise neu ausgegeben wird.

Suchen im »Untergrund«

Ab Zeile 1070 stoßen wir auf ein Unterprogramm. Es enthält zwei Routinen, zum Suchen nach einem bestimmten Bandnamen beziehungsweise einem bestimmten Programmtitel. Diese beiden Auswahlvarianten erscheinen zunächst in Fenster Nummer 2 durch Zeile 1070. Nachdem der Benutzer sich entschieden hat (Zeile 1080), wird der Name abgefragt und, nachdem er in Kleinbuchstaben konvertiert wurde, in n$ abgelegt. Je nach Suchmodus geht es dann in Zeile 1110 oder 1140 weiter. Den ersten Teil, die Suche nach dem Bandnamen, kennen wir bereits von der Druckroutine her. Ab 1140 sehen Sie dann, wie die Suche nach einem Programmtitel abläuft. Wir arbeiten wieder mit einer Doppelschleife, die über alle Bänder und Programme läuft. Jeder Programmtitel wird dabei mit dem Suchbegriff verglichen. Bei einer Übereinstimmung wird dabei ak auf den Schleifenwert i gesetzt. Dadurch erhält man nach der Rückkehr ins Menü »Daten bearbeiten« nun die Datei, die den Suchbegriff enthält, ausgegeben (Zeile 1160). Ansonsten führt eine Fehler-Meldung ohne Änderung von ak zurück (Zeile 1180). Durch Drücken von »2« im Menü »Daten bearbeiten« gelangt man in die Subroutine Blättern. Sie dient dazu, ohne Vorgabe von Bandnamen im Inhaltsverzeichnis herumzustöbern. Beim Vorwärts-Blättern wird das Band mit der Nummer ak+1, beim Rückwärts-Blättern das mit der Nummer ak-1 gezeigt. Bedingung dafür ist natürlich, daß die beschriebenen Bänder überhaupt existieren.

Im Grunde müßten wir zu dieser Funktion lediglich ak ändern, da die komplette Darstellung bereits durch das Untermenü erfolgt. Dies hätte jedoch einen Nachteil im Hinblick auf die Bedienerfreundlichkeit. Will man sich einen Eindruck über den aktuellen Programmbestand verschaffen, so erfordert das normalerweise, einige Bänder zu durchblättern. Bei einer direkten Rückkehr ins Untermenü »Daten bearbeiten« müßte bei jedem weiteren Blättern zuerst wieder die Unterfunktion »Blättern« aufgerufen werden, bevor man dann mit »vorwärts« oder »rückwärts« die Bewegungsrichtung festlegt. Deshalb beinhalten die Zeilen 1280 bis 1310 die Darstellungsroutine aus dem Untermenü noch einmal. Nachdem in den Zeilen 1260 und 1270 das neue ak bestimmt wurde, geben diese Zeilen das zugehörige Inhaltsverzeichnis aus. Danach geht es wieder zurück nach Zeile 1220.

Das nächste Unterprogramm dient zur Änderung des Bandinhaltes einer bereits erfaßten Datei. Diese Funktion macht es möglich, ein neu erfaßtes Programm, das man überschrieben hat, zu löschen. Entsprechend zu diesen Aufgaben gibt es auch einzelne Unterfunktionen. Man erreicht sie aus dem Unterprogramm »Ändern« über den Sprungverteiler in Zeile 1380. Die erste Funktion ist dabei das Löschen eines Eintrages. Zunächst wird gefragt (in Zeile 1430), welche Zeile des Inhaltsverzeichnisses gelöscht werden soll. Die Folgezeile überprüft, ob die angegebene Zahl überhaupt einen erlaubten Wert darstellt. Falls eine nicht existierende Zeilennummer gewählt wurde, so protestiert der CPC und kehrt ins aufrufende Programm zurück. Ansonsten aber beginnt ein munteres Verschiebespiel. Wenn wir eine Zeile löschen wollen, so geschieht dies, indem wir alle höherliegenden Programmtitel dieses Bandes um eine Position im Speicher nach unten verschieben. Der zu löschende Eintrag wird dadurch überschrieben. Nun müssen wir dafür sorgen, daß der Speicherplatz, an dem sich der oberste Programmtitel befand, gelöscht – so spart man Speicherplatz – und die Anzahl der Programmtitel (n(ak,0)) um 1 vermindert wird. Zeile 1450 sorgt für diese Verschiebung. 1460 setzt die Werte richtig, bevor man wieder in das Untermenü zurückgelangt.

Unter der Bezeichnung »Einfügen« finden Sie ab Zeile 1500 die Umkehrung der gerade beschriebenen Funktion. Zuerst muß dabei geprüft werden,

ob noch Speicherplatz zur Verfügung steht. Die Maximalzahl von 15 Titeln darf also noch nicht überschritten sein. Dies übernimmt Zeile 1510. Wenn die Datei noch nicht voll belegt ist, wird festgestellt, ab welchem Titel die Einfügung stattfinden soll.

Nun gibt es zwei Arten. Zum einen kann es sein, daß hier eine Nummer eingegeben wurde, die höher als die bisherige Höchstgrenze liegt. Das kommt vor, wenn beispielsweise ein Programm angehängt wurde. In diesem Fall werden einfach der neue Programmtitel und der zugehörige Zählwerkstand abgefragt und in die ausgewählte Zeile eingetragen (Programmzeile 1550). Sonst erfolgt mit einer Schiebeschleife die Umordnung aller Titel mit höherer Zeilennummer (Zeile 1560), bevor dann auch hier Titel und Zählerstand eingelesen werden. In beiden Fällen ist dann aber immer noch die Gesamtzahl der abgespeicherten Programme zu ändern. Dazu erhöht sich n(ak,0) um 1 (Zeile 1580).

Nun kann es vorkommen, daß man nur den Inhalt einer Zeile ändern will, ohne daß dabei gleich auch noch eine Verschiebung stattfinden muß. Der einfachste Fall ist ein Fehler bei der Eintragung, der korrigiert werden soll. Das Programm verfährt dann folgendermaßen: Die Zeile 1590 fragt die Register, in denen etwas geändert werden soll, ab. Danach wird in 1610 überprüft, ob überhaupt ein Eintrag vorhanden ist. Bei positiver Meldung erfolgen dann in den Zeilen 1620 und 1630 die notwendigen Änderungen.

Ein weiterer Programmpunkt im Menü ist die Funktion »Neuer Dateiname«. Hiermit kann man die Bezeichnung eines Bandes ändern. Bei Bedarf ist es auch möglich, alle alten Dateieinträge zu löschen. Eine einfache Löschschleife (Zeile 1710) leistet dies. Sie schließt auch die Beschreibung des Unterprogrammpunktes »Ändern« ab.

Ändern auf einen Schlag

Lästig ist es immer, mit Einzelfunktionen wie »Ändern« oder »Einfügen«, arbeiten zu müssen. Daher existiert im Hauptmenü noch der Unterpunkt 4. Er ermöglicht es, quasi im Schnelldurchgang, ein ganzes Bandinhaltsverzeichnis auf einen Schlag einzugeben. Ab Zeile 1760 sehen Sie, wie Ihr Schneider dabei vorgehen muß. In Zeile 1780 erhöht sich die Gesamtzahl der gespeicherten Bänder um 1. Voraussetzung dazu ist allerdings, daß noch Speicherplatz verfügbar ist, was aber Zeile 1770 überprüft. Danach wird die Nummer der aktuellen Datei (ak) mit der neuen Obergrenze gleichgesetzt. Dateiname sowie Anzahl und Titel der verschiedenen Dateien mit ihren Zählwerkständen werden abgefragt, ohne daß zwischendurch jeweils eine Rückkehr ins Hauptmenü stattfindet.

Den Schluß unseres Programms bildet eine Ende-Routine. Es stellt sich die Frage, wozu ein solcher Programmteil notwendig ist. Ein einfaches END tut es doch auch! Stellen Sie sich aber einmal vor, was passiert, wenn Sie ohne es zu wollen, im Hauptmenü einfach einmal auf 6 tippen. Und wenn sich Ihr Schneider dann »verabschiedet« und Ihre gerade erst neu erfaßte Datei unwiederbringlich verloren ist, dann wissen Sie, wozu diese Kontroll-Routine dient. Ein vernünftiges Programm sollte alle möglichen Eingabefehler durch solche Kontrollen abfangen, damit der Benutzer nie in die Situation kommt, nicht mehr weiterarbeiten zu können.

Soweit die Erläuterungen zu dem Programm »Band-Kartei«. Es wird Ihnen sicher gute Dienste leisten und für Ordnung sorgen.

Ein Punkt sei aber noch erwähnt. Das gerade beschriebene Programm ist bewußt allgemein gehalten. Es bietet sich also an, es auch für andere Datenverwaltungszwecke zu benutzen. Mit den dazu notwendigen Änderungen haben Sie nach dieser ausführlichen Beschreibung wohl auch keine Probleme mehr. Dafür sorgen auch die Parameter md und ak. So sollten Sie keine Schwierigkeiten haben, aus der Band-Kartei ein Speicherprogramm für Dias oder Musikkassetten zu formen. Hier und bei allen anderen Erweiterungen sind Sie nun selbst gefordert.

(Carsten Straush/hl)

```
10 REM **********************               [74EE]
20 REM ** Initialisierung **                [423C]
30 REM **********************               [FEF2]
40 INK 0,0 : BORDER 0 : INK 1,24: INK 2,
   11: INK 3,6                              [63CC]
50 MODE 0:SPEED WRITE 1                     [DD42]
60 PRINT:PRINT:PRINT                        [5642]
70 PEN 3                                    [647A]
80 PRINT" *****************                 [1B42]
90 PRINT" *{15 SPACE}*"                     [215C]
100 PRINT" *";:PEN 1:PRINT" BAND - KARTE
    I ";:PEN 3: PRINT"*"                    [7004]
110 PRINT" *{15 SPACE}*"                    [A9AE]
120 PRINT" *****************                [B398]
130 PEN 2                                   [36D2]
140 PRINT:PRINT:PRINT:PRINT"von Carsten
    Straush                                 [740C]
150 WHILE i<2000                            [CB6C]
160 z$=INKEY$                               [486C]
170 IF z$<>"" THEN 200                      [0488]
180 i=i+1                                   [0348]
190 WEND                                    [42D0]
200 DEFINT n                                [68F4]
210 DIM n$(20,20):DIM n(20,20)              [51C6]
220 MODE 1:WINDOW#1,1,40,1,3:WINDOW#0,1,
    40,4,22:WINDOW#2,1,40,23,25             [ECE6]
230 PEN#1,1:PEN#0,2:PEN#2,1                 [F6CC]
240 PAPER#1,3:PAPER#0,0:PAPER#2,0           [DF4A]
250 REM ******************                  [AC5E]
260 REM ** Auswahlmenue **                  [1026]
270 REM ******************                  [3062]
280 CLS:CLS#1:CLS#2:PRINT#1:PRINT#1,"{13
     SPACE}Auswahlmenue"                    [FB64]
290 PRINT:PRINT:PRINT"{9 SPACE}DATEI EIN
    LESEN{5 SPACE}(1)"                      [5E6C]
300 PRINT:PRINT"{9 SPACE}DATEI SICHERN{6
     SPACE}(2)"                             [BA82]
310 PRINT:PRINT"{9 SPACE}DATEN BEARBEITE
    N{3 SPACE}(3)"                          [C85A]
320 PRINT:PRINT"{9 SPACE}DATEI EINGEBEN{
    5 SPACE}(4)"                            [35AC]
330 PRINT:PRINT"{9 SPACE}DATEN DRUCKEN{6
     SPACE}(5)"                             [CF98]
340 PRINT:PRINT"{9 SPACE}PROGRAMM BEENDE
    N{3 SPACE}(6)"                          [1498]
350 Z$=INKEY$:IF Z$="" THEN 350 ELSE IF
    ASC(Z$)<49 OR ASC(Z$)>55 THEN 350       [3246]
360 ON VAL(Z$) GOSUB 410,540,930,1760,66
    0,1930                                  [018A]
370 GOTO 280                                [C05A]
380 REM *******************                 [8D0E]
390 REM ** Datei einlesen **                [7904]
400 REM *******************                 [5E00]
410 CLS:CLS#1:PRINT#1:PRINT#1,"{7 SPACE}
    Funktion Datei laden":OPENIN"Bandinh
    alt"                                    [5584]
420 INPUT#9,md                              [CC3E]
430 FOR i= 0 TO md                          [85D0]
440 INPUT#9,n$(i,0):INPUT#9,n(i,0)          [9C9C]
450 FOR j=1 TO n(i,0)                       [43FE]
460 INPUT#9,n$(i,j):INPUT#9,n(i,j)          [4F88]
470 NEXT j:NEXT i                           [720C]
480 CLOSEIN                                 [DB92]
490 ak=1                                    [46EE]
500 RETURN                                  [CE2A]
510 REM *******************                 [7FB0]
520 REM ** Datei sichern **                 [3C2C]
530 REM *******************                 [B9B4]
540 CLS:CLS#1:PRINT#1:PRINT#1,"{8 SPACE}
    Funktion Datei sichern":OPENOUT"Band
    inhalt"                                 [515E]
550 PRINT#9,md                              [6540]
560 FOR i= 0 TO md                          [B4D8]
570 PRINT#9,n$(i,0):PRINT#9,n(i,0)          [3198]
580 FOR j=1 TO n(i,0)                       [7906]
590 PRINT#9,n$(i,j):PRINT#9,n(i,j)          [5484]
600 NEXT j:NEXT i                           [2502]
610 CLOSEOUT                                [4B4A]
620 RETURN                                  [AF30]
630 REM *******************                 [C462]
640 REM ** Druckausgabe **                  [DCEE]
```

Listing. »Band-Kartei«, ein durchschaubares Programm

```
650 REM ******************                           [0066]
660 d=8:CLS:CLS#1:PRINT#1:PRINT#1,"{7 SP
    ACE}Funktion Datei drucken"                     [4110]
670 PRINT:PRINT"Gesamtausdruck (1)":PRIN
    T:PRINT"Einzelnes Band (2)":PRINT:PR
    INT"Verlassen{6 SPACE}(3)"                      [CB74]
680 z$=INKEY$:IF z$="3" THEN RETURN ELSE
     IF z$<>"1" AND z$<>"2" THEN 680                [3934]
690 IF z$="2" THEN 790                              [549C]
700 PRINT:PRINT"Bitte Drucker einschalte
    n!":PRINT"Dann Tastendruck!"                    [BCB6]
710 WHILE INKEY$="":WEND                            [57BC]
720 FOR i=1 TO md                                   [7996]
730 PRINT#d:PRINT#d,n$(i,0)+"{2
    SPACE}"+STR$(n(i,0))+" Programm(e)"             [2B28]
740 PRINT#d                                         [529E]
750 FOR j=1 TO n(i,0)                               [1C04]
760 z$=DEC$((j,"##")+LEFT$("{2 SPACE}"+n
    $(i,j)+SPACE$(20),22)+DEC$((n(i,j),"
    ###"):PRINT#d,z$                                [0F7C]
770 NEXT j:NEXT i                                   [4C12]
780 RETURN                                          [7C3E]
790 PRINT:INPUT"Welches Band (Titel)";n$
    :n$=LOWER$(n$)                                  [D728]
800 FOR i=0 TO md                                   [5592]
810 IF LOWER$(n$(i,0))<>n$ THEN NEXT i              [A6A2]
820 IF i>md THEN PRINT#2,"Existiert nich
    t!":FOR i=1 TO 500:NEXT:RETURN                  [79BA]
830 PRINT:PRINT"Bitte Drucker einschalte
    n!":PRINT"Dann Tastendruck!"                    [13BE]
840 WHILE INKEY$="":WEND                            [9AC4]
850 PRINT#d:PRINT#d,n$(i,0)+"{2
    SPACE}"+STR$(n(i,0))+" Programm(e)"             [6E2E]
860  FOR j=1 TO n(i,0)                              [FA48]
870 z$=DEC$((j,"##")+LEFT$("{2 SPACE}"+n
    $(i,j)+SPACE$(20),22)+DEC$((n(i,j),"
    ###"):PRINT#d,z$                                [6E80]
880 NEXT j                                          [3312]
890 RETURN                                          [AF42]
900 REM ***********************                     [84B2]
910 REM ** Datei bearbeiten **                      [037C]
920 REM ***********************                     [60B6]
930 CLS:CLS#1:CLS#2:PRINT#1:PRINT#1,"{8
    SPACE}Funktion Datei bearbeiten"                [2156]
940 PRINT "Datei Nr.:";ak;" Name:"+n$(ak
    ,0)                                             [62E8]
950 PRINT"akt.PRG's:";n(ak,0)                       [C9B8]
960 FOR i=1 TO n(ak,0)                              [48CE]
970 z$=DEC$((i,"##")+LEFT$("{2 SPACE}"+n
    $(ak,i)+SPACE$(20),22)+DEC$((n(ak,i)
    ,"###")                                         [A9D8]
980 PRINT z$                                        [7D18]
990 NEXT i                                          [3414]
1000 PRINT#2," Suchen (1){2 SPACE}Blaett
     ern (2){2 SPACE}Aendern (3){2 SPACE
     }Verlassen (4)                                 [5DEC]
1010 z$=INKEY$:IF z$="4" THEN RETURN ELS
     E IF z$<>"1" AND z$<>"2" AND z$<>"3
     " THEN 1010                                    [FC0A]
1020 ON VAL(z$) GOSUB 1070,1220,1360                [3BC2]
1030 GOTO 930                                       [56B2]
1040 REM ************                               [4BC2]
1050 REM ** Suchen **                               [3170]
1060 REM ************                               [37C6]
1070 CLS#2:PRINT#2,"Band (1){2 SPACE}Pro
     gramm (2)                                      [FC90]
1080 z$=INKEY$:IF z$<>"1" AND z$<>"2" TH
     EN 1080                                        [246E]
1090 INPUT#2,"Name";n$:n$=LOWER$(n$)                [9D4C]
1100 IF z$="2" THEN 1140                            [8E2E]
1110 FOR i=0 TO md:IF LOWER$(n$(i,0))<>n
     $ THEN NEXT                                    [717A]
1120 IF i>md THEN PRINT#2,"Nicht vorhand
     en!!":FOR i=1 TO 500:NEXT:RETURN               [4618]
1130 ak=i:RETURN                                    [21E2]
1140 FOR i=0 TO md                                  [6BEE]
1150 FOR j=1 TO n(i,0)                              [545A]
1160 IF LOWER$(n$(i,j))=n$ THEN ak=i:RET
     URN                                            [9782]
1170 NEXT j:NEXT i                                  [A468]
1180 PRINT#2,"Nicht vorhanden!!":FOR i=1
      TO 500:NEXT:RETURN                            [13F8]
1190 REM ***************                            [34CA]
1200 REM ** Blaettern **                            [51E0]
1210 REM ***************                            [38BC]
1220 CLS#2:PRINT#2,"{10 SPACE}Hoechste D
     ateinummer";md:PRINT#2,"{4 SPACE}Vo
     r (1){2 SPACE}Zurueck (2){2 SPACE}V
     erlassen (3)"                                  [4614]
1230 z$=INKEY$:IF z$<>"1" AND z$<>"2" AN
     D z$<>"3" THEN 1230                            [7DA6]
1240 IF z$="3" THEN RETURN                          [E36E]
1250 CLS                                            [0494]
1260 IF z$="2" THEN ak=MAX(ak-1,1)                  [243E]
1270 IF z$="1" THEN ak=MIN(ak+1,md)                 [0176]
1280 CLS:PRINT "Datei Nr.:";ak;" Name:"+
     n$(ak,0):PRINT"akt.PRG's:";n(ak,0)             [852C]
1290 FOR i=1 TO n(ak,0)                             [CC28]
1300 z$=DEC$((i,"##")+LEFT$("{2 SPACE}"+
     n$(ak,i)+SPACE$(20),22)+DEC$((n(ak,
     i),"###") :PRINT z$                            [EB6A]
1310 NEXT i                                         [865A]
1320 CLS#2:GOTO 1220                                [72EA]
1330 REM *************                              [C11A]
1340 REM ** Aendern **                              [D922]
1350 REM *************                              [F71E]
1360 CLS#2:PRINT#2,"Loeschen (1){2 SPACE
     }Einfuegen (2){2 SPACE}Aendern (3)"
     :PRINT#2,"Neuer Dateiname (4)"                 [87E4]
1370 z$=INKEY$:IF z$<>"1" AND z$<>"2" AN
     D z$<>"3" AND z$<>"4" THEN 1370                [4C00]
1380 CLS#2:ON VAL(z$) GOSUB 1430,1510,15
     90,1670                                        [44B8]
1390 RETURN                                         [8A9A]
1400 REM **************                             [CF6A]
1410 REM ** Loeschen **                             [FB06]
1420 REM **************                             [0B6E]
1430 INPUT#2,"Welche Zeile";z                       [FEC6]
1440 IF z>n(ak,0) OR n(ak,0)=0 THEN PRIN
     T#2,"Unmoeglich!!":FOR i=1 TO 500:N
     EXT:RETURN                                     [803A]
1450 FOR i=z TO n(ak,0):n$(ak,i)=n$(ak,i
     +1):n(ak,i)=n(ak,i+1):NEXT i                   [B596]
1460 n$(ak,n(ak,0))="":n(ak,n(ak,0))=0:n
     (ak,0)=n(ak,0)-1                               [AD2C]
1470 RETURN                                         [8F98]
1480 REM ***************                            [8FCE]
1490 REM ** Einfuegen **                            [69E0]
1500 REM ***************                            [39C0]
1510 IF n(ak,0)=15 THEN PRINT#2,"Datei i
     st voll!!":FOR i=1 TO 500:NEXT:RETU
     RN                                             [28F4]
1520 INPUT#2,"Welche Zeile (1-15){2 SPAC
     E}0=Verlassen";z                               [08F0]
1530 IF z=0 THEN RETURN                             [149C]
1540 IF z>15 OR z<1 THEN 1520                       [476C]
1550 IF z>n(ak,0) THEN n(ak,0)=z:INPUT#2
     ,"Name des Programms";n$(ak,z):INPU
     T#2,"Zaehlerstand";n(ak,z):RETURN              [5F28]
1560 FOR i=n(ak,0) TO z STEP-1:n$(ak,i+1
     )=n$(ak,i):n(ak,i+1)=n(ak,i):NEXT i
                                                    [980E]
1570 INPUT#2,"Name des Programms";n$(ak,
     z):INPUT#2,"Zaehlerstand";n(ak,z)              [BFCE]
1580 n(ak,0)=n(ak,0)+1:RETURN                       [FFDE]
1590 CLS#2:INPUT#2,"Welche Zeile 0=verla
     ssen";z                                        [6276]
1600 IF z=0 THEN RETURN                             [F798]
1610 IF z<1 OR z>n(ak,0) THEN PRINT#2,"K
     ein Eintrag!!":FOR i=1 TO 500:NEXT:
     RETURN                                         [A32A]
1620 INPUT#2,"Neuer Programmname";n$(ak,
     z)                                             [A9A6]
1630 INPUT#2,"Neuer Zaehlerstand";n(ak,z
     ):RETURN                                       [B892]
1640 REM **********************                     [F0C2]
1650 REM ** Neuer Dateiname **                      [E9FE]
1660 REM **********************                     [2AC6]
1670 INPUT#2,"Neuer Dateiname";n$(ak,0):
     n$(ak,0)=LEFT$(n$(ak,0),15)                    [0EDE]
1680 PRINT#2,"Datei loeschen j/n"                   [4D80]
1690 z$=LOWER$(INKEY$):IF z$="n" THEN RE
     TURN                                           [EA64]
1700 IF z$<>"j" THEN 1690                           [6E38]
1710 FOR i=1 TO 15:n$(ak,i)="":n(ak,i)=0
     :NEXT:n(ak,0)=0                                [A0BA]
1720 RETURN                                         [B094]
1730 REM ********************                       [EB6E]
1740 REM ** Datei eingeben **                       [3538]
1750 REM ********************                       [C772]
1760 CLS#1:PRINT#1:PRINT#1,"{12 SPACE}Da
     tei eingeben"                                  [E91C]
1770 IF md=20 THEN PRINT#2," Dateibestan
     d erschoepft!!":FOR i=1 TO 1000:NEX
     T:RETURN                                       [BC36]
1780 md=md+1:ak=md                                  [CB7E]
1790 IF n=-1 THEN RETURN                            [33F0]
1800 CLS:PRINT "Datei Nr.:";ak                      [9558]
1810 INPUT#2,"Dateiname V=Verlassen";n$             [89F4]
1820 n$=LEFT$(n$,15)                                [15FC]
1830 IF LOWER$(n$)="v" THEN md=md-1:ak=m
     d:RETURN                                       [3DF8]
1840 n$(ak,0)=n$                                    [538E]
1850 INPUT#2,"Wieviele Programme";n                 [77E0]
1860 n(ak,0)=n                                      [6202]
1870 FOR i= 1 TO n                                  [837E]
1880 PRINT#2,"Programmname "+STR$(i);:IN
     PUT#2,n$(ak,i):INPUT#2,"Zaehlerstan
     d";n(ak,i)                                     [F6C8]
1890 NEXT:RETURN                                    [FA96]
1900 REM **********************                     [10C0]
1910 REM ** Programmabbruch **                      [16E6]
1920 REM **********************                     [2AC4]
1930 BORDER 6,24:PRINT#2,"Sie haben Prog
     rammabbruch gewaehlt. Solldas Progr
     amm wirklich abgebrochen werdenj/n?
     "                                              [4610]
1940 z$=LOWER$(INKEY$):IF z$="j" THEN BO
     RDER 0:END ELSE IF z$<>"n" THEN 194
     0 ELSE BORDER 0:RETURN                         [7636]
```

Listing. »Band-Kartei«, ein durchschaubares Programm (Schluß)

Schwertransport

Besitzer eines Schneider CPC 464, die ihren »Maschinenpark« um ein Diskettenlaufwerk erweitert haben, stehen sehr bald vor einem Problem: Wie läßt sich die vorhandene Software von Kassette auf Diskette übertragen?

Noch heute (und voraussichtlich auch in Zukunft) ist die meiste Spielsoftware ausschließlich auf Kassette erhältlich.

Programme, die Kassettensoftware auf Diskette bringen, wie zum Beispiel »Transmat«, besitzt nicht jeder. Dazu kommt, daß sie nur einen Teil der Software transferieren und das oft auch nur mit erheblichen Eingriffen des Benutzers in die kopierten Programme. Das heißt natürlich nicht, daß hier eine Anleitung für jede erdenkliche Software gegeben werden kann; dazu gibt es zu unterschiedliche Konzepte. Viele Programme haben sehr spezielle Laderoutinen, wie zum Beispiel das Flashloading (bei »Sorcery« und anderen Titeln), oder einfach ein anderes Ladeformat (zum Beispiel »Spy vs Spy«). Auf Grund der Komplexität können wir an dieser Stelle darauf nicht eingehen.

Zunächst wollen wir einen Blick auf Software werfen, die ganz »ordinär« blockweise im Standardformat (»SAVE "name",p«) gespeichert ist. Für den Transfer benötigen Sie einige Informationen über das Programm. So prüfen Sie erst einmal mit dem CAT-Befehl, in wieviele Teile sich das Programm gliedert (Laderoutine, Titelbild, Hauptprogramm, etc.).

Nehmen wir beispielsweise an, das Programm besteht aus zwei Teilen: zum einen aus dem, natürlich geschützten, Basic-Programm und einem anschließenden Maschinensprache-Teil. Den »Schutz« des Basic-Programms müssen Sie aus zwei Gründen entfernen: Erstens läßt es sich nicht mit einfachem »LOAD« und anschließendem »SAVE« übertragen. Dazu kommmt, daß Sie meist kleine Eingriffe vornehmen müssen, deren Zweck Sie später erkennen werden. Den Listschutz entfernen Sie mit Hilfe eines kleinen Programms. Für den CPC 464 benötigen Sie Listing 1, für die CPC 664/6128 Listing 2. Der gelistete Basic-Teil sieht wie folgt aus: Auf den Aufbau des Titelbildes und die Farbdefinition folgt der Ladebefehl und schließlich der Aufruf des Maschinencode-Programms mit »CALL adresse«. Von besonderem Interesse ist für uns die »Laderoutine«. Sie besteht in aller Regel nur aus der Anweisung »LOAD "!"« und lädt das auf der Kassette folgende Programm.

Wie Sie vielleicht wissen, müssen Sie das Ausrufezeichen durch einen Programmnamen ersetzen, der die Bedingungen des Disketten-Betriebssystems erfüllt (acht Zeichen, eventuell mit einer Erweiterung von drei Zeichen getrennt durch einen Punkt). So ist der Basic-Lader bereits aufbereitet und Sie speichern ihn auf Diskette.

Die Übertragung des Maschinencodes kostet ungleich mehr Mühe: Sie müssen seine Ladeadresse und Länge herausfinden. Lädt und startet der Basic-Teil das Maschinenprogramm nicht über »LOAD "!":CALL adresse«, sondern mit »RUN"!"«, benötigen Sie noch zusätzlich die Startadresse. Hilfe finden Sie in Listing 3. Diese Hilfsroutine liest die entsprechenden Informationen aus dem sogenannten Header des jeweiligen Speicherblocks.

In unserem Beispiel ist die Ladeadresse 40000 und der Maschinencode 1500 Byte lang. Geben Sie nun folgende Basic-Zeilen ein:

```
10 OPENOUT "DUMMY":
   MEMORY 40000-1:CLOSEOUT
20 |TAPE.IN : |DISC.OUT
30 LOAD ""
40 SAVE "name",b,40000,1500
```

Zeile 10 richtet mit dem OPENOUT-Befehl einen Pufferbereich von 4096 Byte ein. Das ist wichtig, denn wenn der Maschinencode ab einer Adresse unterhalb 5000 beginnt, erscheint bei fehlendem Puffer und »LOAD« nur noch die Fehlermeldung »MEMORY FULL«. Der MEMORY-Befehl setzt den HIMEM-Zeiger herab und reserviert den nachfolgenden Speicherbereich für den Maschinencode. Zeile 20 kanalisiert den Datentransfer in die gewünschte Richtung. Zeile 30 lädt das nächste verfügbare Programm von der Kassette. Die Auswahl der richtigen Datei treffen Sie, indem Sie das Band vorher an die entsprechende Stelle spulen. Zeile 40 schließlich speichert das Programm auf Diskette. Mit diesem einfachen Rezept

Fortsetzung auf Seite 50

```
10 ' Aufhebung Listschutz CPC 464          [E6DC]
20 POKE &AC01,&C3:POKE &AC02,&90,POKE &A
   C03,&C0                                 [7582]
```

Listing 1. Aufheben des Listschutzes beim CPC 464

```
10 ' Aufhebung Listschutz CPC 664/6128   [ABE0]
20 FOR i=&BB9F TO &BBA4 : READ a : POKE
   i,a : NEXT                            [12D2]
30 DATA &3E,0,&32,&2C,&AE,&C9            [64B6]
```

Listing 2. Auch beim CPC 664/6128 Programme wieder »sichtbar«

```
10 ' ****************
      * Headerreader *
       ****************                  [3EB0]
20 INK 0,0:BORDER 0:INK 1,24:INK 2,18:IN
   K 3,15                                [50B6]
30 ON ERROR GOTO 300                     [89AC]
40 MODE 1:|TAPE 'Umschalten auf Cassette
                                         [75DA]
50 filename$=""                          [DDD6]
60 x=&B88C:OPENIN ""                     [937A]
70 ' Wie heisst das File?                [D8B4]
80 FOR i=x TO x+15                       [4132]
90 a=PEEK(i):IF a=0 THEN a=32            [0E5E]
100 filename$=filename$+CHR$(a):NEXT     [B81E]
110 t=PEEK(x+18) 'Welches Programm       [93B8]
120 l=PEEK(x+24)+256*PEEK(x+25) 'Laenge  [72C0]
130 a=PEEK(x+21)+256*PEEK(x+22) 'Anfang  [B99E]
140 s=PEEK(x+26)+256*PEEK(x+27) 'Start   [E35E]
150 ' Alle Daten anzeigen                [31F0]
160 CLS                                  [8332]
170 PRINT"{CTRL O}{CTRL A}Programmname{2
    SPACE}: {CTRL O}{CTRL B}";filename$
                                         [0894]
180 PRINT"{CTRL J}{CTRL O}{CTRL A}Progra
   mmart{3 SPACE}: {CTRL O}{CTRL B}";    [466C]
190 IF t<2 THEN PRINT"BASIC";            [4D50]
200 IF t=2 OR t=3 THEN PRINT"BINAER";    [556A]
210 IF t=1 OR t=3 THEN PRINT" GESCHUETZT
    ";                                   [6A54]
220 PRINT                                [4582]
230 PRINT"{CTRL J}{CTRL O}{CTRL A}Anfang
    sadresse: {CTRL O}{CTRL B}&";HEX$(a,
    4);USING "{3 SPACE}{CTRL O}{CTRL C}
    #####";a                             [9E5C]
240 PRINT"{CTRL J}{CTRL O}{CTRL A}Progra
    mmlaenge: {CTRL O}{CTRL B}&";HEX$(l,
    4);USING "{3 SPACE}{CTRL O}{CTRL C}
    #####";l                             [CEA2]
250 PRINT"{CTRL J}{CTRL O}{CTRL A}Starta
    dresse{2 SPACE}: {CTRL O}{CTRL B}&";
    HEX$(s,4);USING "{3 SPACE}{CTRL O}{C
    TRL C} #####";s                      [B1C8]
260 PRINT"{CTRL O}{CTRL A}":CLOSEIN      [C7C0]
270 PRINT"{CTRL J}{CTRL J}{CTRL J}Weiter
    es Programm untersuchen? (J/N)"      [F540]
280 WHILE a$<>"J" AND a$<>"N":a$=UPPER$(
    INKEY$):WEND                         [881A]
290 IF a$="J" THEN RUN ELSE CLS:END      [A5F4]
300 RESUME NEXT 'Fehlerbehandlung aus    [9C02]
```

Listing 3. Im Programmkopf stehen wichtige Informationen

Großes Preisausschreiben

Ihre Meinung ist gefragt!

Erfüllt dieses Sonderheft Ihre Anforderungen? Haben Sie es gerne gelesen oder bereuen Sie den Kauf? Damit wir uns besser auf Ihre Wünsche einstellen können, schicken Sie uns bitte den Fragebogen zurück und gewinnen Sie mit etwas Glück einen der wertvollen Preise.

- **1 Diskettenlaufwerk FX-1 gestiftet von Vortex**
- **1 Speicherkarte SP 512 für CPC 464 oder 664 von Vortex**
- **3 CP/M-Programme (Wordstar, dBase II oder Multiplan) freier Wahl**
- **10 Gutscheine über ein Buch freier Wahl aus dem Markt & Technik Verlag**
- **10 T-Shirts im »Happy-Computer«-Look**

Um an unserer Verlosung teilzunehmen, müssen Sie nur den folgenden Fragebogen ausfüllen und bis zum 31. Juli 1986 an uns zurückschicken. Unsere Adresse:

Redaktion Happy-Computer
Markt & Technik Verlag AG
Kennwort: Schneider
Hans-Pinsel-Straße 2
8013 Haar bei München
Der Rechtsweg ist ausgeschlossen.

Fragebogen zum Schneider-Sonderheft von Happy-Computer

Wie hat Ihnen dieses Sonderheft insgesamt gefallen?

☐ sehr gut ☐ gut
☐ mittel ☐ weniger gut
☐ gar nicht ______

Wie fanden Sie die einzelnen Rubriken?

	sehr interessant	interessant	weniger interessant
Hardware	☐	☐	☐
Software	☐	☐	☐
Basteleien	☐	☐	☐
Grundlagen	☐	☐	☐
Joyce-Teil	☐	☐	☐
Einsteiger-Teil	☐	☐	☐
CP/M-Teil	☐	☐	☐
Anwendungs-Listings	☐	☐	☐
Spiele-Listings	☐	☐	☐
Tricks & Tips	☐	☐	☐
CP/M-Tips	☐	☐	☐

Welche Beiträge haben Ihnen am besten gefallen?

1. ______
2. ______
3. ______
4. ______
5. ______

Welche Beiträge haben Ihnen gar nicht gefallen?

1. ______
2. ______
3. ______

4. ______________________

5. ______________________

Welche Themen sollen in weiteren Sonderheften behandelt werden?

1. ______________________
2. ______________________
3. ______________________
4. ______________________
5. ______________________

Welche Themen sollen in Happy-Computer ausführlich behandelt werden?

1. ______________________
2. ______________________
3. ______________________
4. ______________________
5. ______________________

Wie oft würden Sie sich ein Sonderheft oder eine spezielle Schneider-Zeitschrift von Markt & Technik kaufen?

☐ monatlich ☐ zweimonatlich

☐ vierteljährlich ☐ halbjährlich

Wie teuer darf Ihrer Meinung nach ein spezielles Schneiderheft sein?

Sonderheft ____ Mark Zeitschrift ____ Mark

Welche Computer-Zeitschriften lesen Sie regelmäßig?

	Kaufen	Lesen
1.		
2.		
3.		
4.		
5.		

Welche Schneider-Sonderhefte von Happy-Computer haben Sie sich schon gekauft?

☐ 1. Schneider-Sonderheft (SH 2/85)

☐ 2. Schneider-Sonderheft (SH 1/86)

☐ 3. Schneider-Sonderheft (SH 4/86)

Welchen Computer besitzen Sie?

☐ CPC 464 ☐ CPC 664

☐ CPC 6128 ☐ Joyce

☐ einen anderen, welchen? ______________________

Haben Sie vor, sich 1986 einen anderen Computer zu kaufen?

☐ ja ☐ nein

☐ Wenn ja, welchen? ______________________

☐ ein anderes Gerät von Schneider, welchen? ______________________

☐ Atari ST ☐ Commodore 64/128

☐ Commodore Amiga ☐ MS-DOS-Computer

____

Welche Peripherie besitzen Sie für Ihren Schneider-Computer?

☐ Diskettenlaufwerk, Type? ______________________

☐ Zweitlaufwerk, Type? ______________________

☐ Speichererweiterung, Type? ______________________

☐ Drucker, Type? ______________________

☐ Modem, Type? ______________________

☐ RS232-Schnittstelle, Type? ______________________

☐ Sonstiges? ______________________

____

Welche Peripherie wollen Sie sich demnächst noch kaufen?

☐ Diskettenlaufwerk, Type? ______________________

☐ Zweitlaufwerk, Type? ______________________

☐ Festplatte, Type? ______________________

☐ Speichererweiterung, Type? ______________________

☐ Drucker, Type? ______________________

☐ Modem, Type? ______________________

☐ RS232-Schnittstelle, Type? ______________________

☐ Sonstiges? ______________________

____

Welches Peripheriegerät würden Sie, falls Sie gewinnen, bevorzugen?

☐ FX-1-Station von Vortex

☐ Speichererweiterung für 464/664 von Vortex

Ich bin damit einverstanden, daß die hier gemachten Angaben elektronisch verarbeitet werden.

Anschrift:

Name/Vorname ______________________

Straße ______________________

PLZ/Ort ______________________

Alter ____ Jahre ______________________

JETZT AUF SCHNEIDER-COMPUTERN:



DIE PROGRAMM-BIBLIOTHEK FÜR TURBO PASCAL®

Markt&Technik
Schneider CPC Software
Turbo Lader
Die Programm-Bibliothek für Turbo Pascal
über 100 Prozeduren und Funktionen in Turbo Pascal Source Code: Bitmanipulation, Sortierverfahren, Splinefunktionen, Fouriertransformation, Regressionsanalyse und vieles mehr.
3" Schneider-Format

TURBO-Lader-Grundpaket

Das TURBO-Lader-Grundmodul ist eine umfangreiche Programm-Bibliothek für den TURBO-Pascal-Programmierer. Sie umfaßt zahlreiche ausführlich dokumentierte Prozeduren und Funktionen, die der Profi zur schnellen Lösung seiner Programmieraufgaben verwenden kann und dem Einsteiger das Erlernen der Pascal-Programmierung erleichtern. Das Grundpaket TURBO-Lader bietet ein breitgefächertes Spektrum von Routinen, beginnend bei Bitmanipulation über optimierte Sortierverfahren bis hin zur Anwendung von Splinefunktionen, Fouriertransformation und Regressionsanalyse. Des weiteren Disketten-Routinen zum Lesen eines Inhaltsverzeichnisses oder zum Lesen und Schreiben einzelner Sektoren, Routinen zur Datenüberprüfung, ein Spooler mit Steuerroutinen, erweiterte Stringverarbeitung und vieles mehr. Alle Routinen werden im kommentierten Quellcode für den TURBO-Pascal-Compiler ausgeliefert.

Das TURBO-Lader-Grundpaket erfordert den TURBO-Pascal-Compiler. Es ist lieferbar auf 3"- und 5 1/4"-Disketten und lauffähig auf dem Schneider CPC 464, CPC 664, CPC 6128 und Joyce.

3"-Disk. Best.-Nr. MS 413
5¼"-Disk. Best.-Nr. MS 415

DM 138,–/sFr. 125,–/öS 1380,–*

Markt&Technik
Schneider CPC Software
Turbo Lader Business
Die Programm-Bibliothek für Turbo Pascal
Ein komfortabler Bildschirm-Maskengenerator und eine professionelle Dateiverwaltung in Turbo Pascal Source Code.
3" Schneider-Format

TURBO-Lader Business

TURBO-Lader Business umfaßt einen komfortablen Bildschirm-Maskengenerator und eine professionelle Dateiverwaltung. Der Maskengenerator gibt dem Pascal-Programmierer ein Werkzeug zur einfachen Bearbeitung von Bildschirm-Masken in die Hand. Eine Maske kann beliebig viele Textfelder, bis zu 128 Eingabe- und 128 Ausgabefelder enthalten. Eingabefelder können auf komfortable Art editiert und auf Gültigkeit überprüft werden. Das Dateiverwaltungsmodul unterstützt die Programmierung von Datenbankanwendungen und Stammdatenverwaltungen. Es besteht aus einer komfortablen Datensatz- und Indexverwaltung mit mehreren Schlüsseln und Index-Dateien, die einen sekundenschnellen Zugriff auf beliebige Daten ermöglicht. Mit diesen beiden Modulen stehen dem Anwendungsprogrammierer zwei professionelle Werkzeuge zur zeit- und kostensparenden Erstellung kommerzieller Anwendungen zur Verfügung. Alle Routinen werden im kommentierten Quellcode für den TURBO-Pascal-Compiler ausgeliefert.

TURBO-Lader Business erfordert den TURBO-Pascal-Compiler und das TURBO-Lader-Grundpaket. Es ist lieferbar auf 3"- und 5 1/4"-Disketten und lauffähig auf dem Schneider CPC 464, CPC 664, CPC 6128 und Joyce.

3"-Disk. Best.-Nr. MS 423
5¼"-Disk. Best.-Nr. MS 425

DM 148,–/sFr. 132,–/öS 1480,–*

Markt&Technik
Schneider CPC Software
Turbo Lader Science
Die Programm-Bibliothek für Turbo Pascal
Technisch-wissenschaftliche Funktionen und professionelle statistische Methoden für die Bereiche Medizin, Betriebs- und Volkswirtschaft, Technik und Naturwissenschaften in Turbo Pascal Source Code.
3" Schneider-Format

TURBO-Lader Science

TURBO-Lader Science ist eine Sammlung technisch/wissenschaftlicher Funktionen und professioneller statistischer Verfahren für die Bereiche Medizin, Betriebs- und Volkswirtschaft, Technik und Naturwissenschaften. Das Modul enthält alle arithmetischen Operationen zur Verarbeitung komplexer Variablen inklusive der Umrechnung der Darstellung und die wichtigsten komplexen Funktionen wie Potenz, Wurzel, trigonometrische, transzendente und exponentielle Funktionen. Darüber hinaus ist ein vollständiges Paket zur Verarbeitung komplexer Matrizen und Vektoren enthalten. Der Statistikteil ist ein praktisches und direkt verwendbares Werkzeug zur computerunterstützten, effektiven Datenanalyse. Er umfaßt eine Vielzahl statistischer Funktionen mit den Schwerpunkten Regression und Korrelation, deskriptive Statistik, Faktoranalyse und Testverfahren. Alle Routinen werden im kommentierten Quellcode für den TURBO-Pascal-Compiler ausgeliefert.

TURBO-Lader Science erfordert den TURBO-Pascal-Compiler und das TURBO-Lader-Grundpaket. Es ist lieferbar auf 3"- und 5 1/4"-Disketten und lauffähig auf dem Schneider CPC 464, CPC 664, CPC 6128 und Joyce.

3"-Disk. Best.-Nr. MS 433
5¼"-Disk. Best.-Nr. MS 435

DM 189,–/sFr. 169,–/öS 1890,–*

*inkl. MwSt., unverbindliche Preisempfehlung.

Übrigens können Sie auch alle TURBO-Pascal-Produkte für Schneider CPC 464/664/6128 und Joyce bei Markt & Technik beziehen:

- TURBO Pascal 3.0, Best.-Nr. MS 514 (CPC), Best.-Nr. MS 515 (Joyce) DM 225,72*
- TURBO Pascal 3.0 mit Grafikunterstützung, Best.-Nr. MS 524 (CPC) DM 285,–*
- TURBO Tutor (deutsch), Best.-Nr. MS 534 (CPC), Best.-Nr. MS 535 (Joyce) DM 104,86*
- TURBO Tutor (englisch), Best.-Nr. MS 544 (CPC), Best.-Nr. MS 545 (Joyce) DM 104,86*
- TURBO Graphix Toolbox, Best.-Nr. MS 564 (CPC) DM 225,72*
- TURBO Toolbox, Best.-Nr. MS 554 (CPC), Best.-Nr. MS 555 (Joyce), DM 225,72*

TURBO-Pascal® ist ein Warenzeichen der Borland Inc., USA. TURBO-Lader, TURBO-Lader Business und TURBO-Lader Science sind Warenzeichen der Fa. Lauer & Wallnitz.

Diese Markt & Technik-Softwareprodukte erhalten Sie in den Fachabteilungen der Kaufhäuser und in Computershops.

Wenn Sie direkt beim Markt & Technik Verlag bestellen wollen:
Nur gegen Vorauskasse, Verrechnungsscheck oder mit der eingedruckten Zahlkarte.



Unternehmensbereich Buchverlag
Hans-Pinsel-Straße 2, 8013 Haar bei München

Bestellungen im Ausland bitte an untenstehende Adressen:

Schweiz: Markt & Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel. 042/41 56 56

Österreich: Ueberreuter Media Handels- und Verlagsges. mbH, Alser Str. 24, A-1091 Wien, Tel. 0222/48 15 38-0

```
10 FOR i=2000 TO 2027                          [F03C]
20 READ a : POKE i,a                           [81BC]
30 NEXT : SAVE "move.bin",b,2000,28            [93FE]
40 DATA 221,78,2,221,70,3,221,110,4,221,
   102,5,229,175,237,82,225,56,0,235,9,2
   35,27,9,43,237,184,201                      [5C6A]
```

Listing 4. Eine Verschieberoutine hält das DOS-RAM frei

```
10 '* Transferprogramm fuer headerlose *
     * Files von Cassette auf Diskette
   . *                                         [0E62]
20 '*    Das Programm ist NICHT fuer    *
     *        RAUBKOPIEN gedacht!!!!!
     *                                         [C2D6]
30 '*       (c) 1986 by ms-Software     *      [B5FA]
40 MODE 1                                      [11F4]
50 a$=""                                       [9316]
60 FOR n=1 TO 12:READ x:a$=a$+CHR$(x):NE
   XT                                          [C418]
70 INPUT "{CTRL O}{CTRL A}Anfangsadresse
   : {CTRL O}{CTRL B}",an:x$=HEX$(an,4):
   MID$(a$,2,2)=CHR$(VAL("&"+RIGHT$(x$,2
   )))+CHR$(VAL("&"+LEFT$(x$,2)))              [9042]
80 INPUT "{CTRL O}{CTRL A}Laenge: {CTRL
   O}{CTRL B}",la:x$=HEX$(la,4):MID$(a$,
   4,2)=CHR$(VAL("&"+RIGHT$(x$,2)))+CHR$
   (VAL("&"+LEFT$(x$,2)))                      [768E]
90 INPUT "{CTRL O}{CTRL A}Satzkennzeiche
   n: {CTRL O}{CTRL B}",sa:x$=HEX$(sa,2)
   :MID$(a$,8,1)=CHR$(VAL("&"+x$))             [A31C]
100 IF an<0 THEN an=an+2^16                    [2E0E]
110 IF an>HIMEM THEN 120 ELSE OPENOUT "d
    ummy":MEMORY an-1:CLOSEOUT                 [6616]
120 fr=FRE(""):CLS:a=PEEK(@a$+1)+256*PEE
    K(@a$+2):CALL a                            [F84C]
130 PRINT"{CTRL J}{CTRL J}{CTRL J}{CTRL
    O}{CTRL A}Soll ich das Programm auf
    Diskette{6 SPACE}speichern? (J/N)"         [AB1E]
140 WHILE a$<>"J" AND a$<>"N":a$=UPPER$(
    INKEY$):WEND                               [5510]
150 IF a$="N" THEN CLS:END ELSE INPUT "{
    CTRL J}{CTRL J}Bitte Dateiname einge
    ben:{CTRL O}{CTRL B} ",a$                  [394A]
160 SAVE ""+a$,b,an,la+1:END                   [BAD2]
170 DATA &21,&00,&00                           [9FBE]
180 ' LD HL,&xxxx    Anfangsadresse            [F4F6]
190 DATA &11,&00,&00                           [BAC0]
200 ' LD DE,&xxxx    Laenge                    [8420]
210 DATA &3e,&00                               [A6BA]
220 ' LD A,&xx       Satzkennzeichen           [C244]
230 DATA &cd,&a1,&bc                           [24AE]
240 ' CALL &BCA1     CAS READ                  [70D2]
250 DATA &c9                                   [6F66]
260 ' RET            Zurueck ins BASIC         [833E]
```

Listing 5. Auch »kopflose« Kassetten-Dateien sind kein Problem

```
10 ' *****************************
      * Screensave in einem Block *
      *****************************            [82BE]
20 MODE 2:SPEED WRITE 1                        [0140]
30 DEG                                         [C2A6]
40 FOR n=0 TO 360 STEP 2                       [A820]
50 MOVE 320+200*SIN(n*2),200+200*SIN(n):
   DRAW 320-200*COS(n),200-200*SIN(n):NE
   XT                                          [3C3C]
60 MEMORY 39999                                [BB2C]
70 FOR n=40000 TO 40011                        [B10C]
80 READ x:POKE n,x:NEXT                        [4BA0]
90 CALL 40000:END                              [3D94]
100 DATA &21,&00,&c0                           [0716]
110 ' LD HL,&C000    Anfangsadresse            [88CE]
120 DATA &11,&00,&40                           [C8BA]
130 ' LD DE,&4000    Laenge                    [4EEC]
140 DATA &3e,&01                               [A1C0]
150 ' LD A,&01       Satzkennzeichen           [B52A]
160 DATA &cd,&9e,&bc                           [69CA]
170 ' CALL &BC9E     CAS WRITE                 [70CC]
180 DATA &c9                                   [AC6A]
190 ' RET            Zurueck ins BASIC         [9E42]
```

Listing 6. Speichern Sie beliebige Binär-Dateien in einem »Rutsch«

```
10 ' *****************************
      * Screenload in einem Block *
      *****************************            [0FA0]
20 MODE 2                                      [DFF2]
30 MEMORY 39999                                [1B26]
40 FOR n=40000 TO 40011                        [1706]
50 READ x:POKE n,x:NEXT                        [1D9A]
60 CALL 40000                                  [AD6C]
70 GOTO 70                                     [268E]
80 ' oder auch :DISC:SAVE "SCREEN",b,&c0
   00,&4000                                    [C95A]
90 DATA &21,&00,&c0                            [28C6]
100 ' LD HL,&C000    Anfangsadresse            [F8CC]
110 DATA &11,&00,&40                           [EEB8]
120 ' LD DE,&4000    Laenge                    [6CEA]
130 DATA &3e,&01                               [DFBE]
140 ' LD A,&01       Satzkennzeichen           [4228]
150 DATA &cd,&a1,&bc                           [19B0]
160 ' CALL &BCA1     CAS READ                  [41D4]
170 DATA &c9                                   [1A68]
180 ' RET            Zurueck ins BASIC         [FF40]
```

Listing 7. Ladevorgänge von Kassette in fast der halben Zeit

lassen sich viele der gängigen Programme auf Diskette »schaufeln«.

Schwieriger wird es, wenn der Maschinencode eines Programms einen Teil des RAM-Speichers belegt, der vom Disketten-Betriebssystem (DOS) reserviert ist (bei AMSDOS sind das 1280 Byte ab Adresse 42620). Wenn Sie ihn normal zu laden versuchen, bricht der CPC beim Überschreiben dieses reservierten Bereichs ab und meldet »BAD COMMAND«. Also müssen Sie den Maschinencode in einen niedrigeren Speicherbereich laden, um ihn dann an die ursprünglich vorgesehene Adresse zu verschieben. Den Maschinencode-Profis unter Ihnen fällt hierzu natürlich sofort das Stichwort »LDDR« ein. Der Z80-Prozessor stellt für so eine »Schiebung« nämlich zwei mächtige Maschinenbefehle zur Verfügung: »LDIR« und »LDDR«. Aus Basic sind diese Befehle jedoch nicht direkt greifbar. Deshalb müssen Sie sich mit einer Basic-Routine (Listing 4) helfen, die den notwendigen Maschinencode in den Speicher POKEt. Die erzeugte Maschinenroutine ist im Speicher frei verschiebbar. Solange noch ein paar Byte für Basic freibleiben, können Sie folglich das Maschinenprogramm so tief herunterladen, wie Sie wollen (oder müssen).

Unser zweiter fiktiver Transfer-Kandidat hat eine Länge von 40 KByte und belegt normalerweise den Speicherbereich von Adresse 3000 bis 43959. Um das DOS-RAM freizuhalten, genügt es, die neue Ladeadresse auf etwa 1500 zu verlegen. Die 28 Byte der Verschieberoutine legen wir darunter, deshalb die Befehlsfolge »OPENOUT "dummy" : MEMORY 1500-28 : CLOSEOUT«. Laden Sie nun zuerst von Diskette mit »LOAD "MOVE.BIN", 1500-28«. Dann schalten Sie auf Kassette um »|TAPE.IN« (oder bei VDOS mit »|CAS.IN«) und laden mit »LOAD "name",1500« das Hauptprogramm. Der Aufruf »CALL 1500-28,1500, 40960,3000« verschiebt dann den Maschinencode auf die richtige Adresse. Die Argumente zum CALL-Befehl bedeuten in der Reihenfolge von links nach rechts: Einsprung der Verschieberoutine, Basisadresse, Länge und neue Basis des zu verschiebenden Programms.

Eine weitere Steigerung des Schwierigkeitsgrades bedeutet es, wenn das Programm, das Sie kopieren wollen, den Speicherbereich von unten an belegt, in dem normalerweise die Basic-Programme residieren. So ist beispielsweise das Spiel »3D-Grand-Prix« nur ab Adresse 40 hex lauffähig. In solchen Fällen müssen Sie sich ein Transferprogramm in purem Maschinencode schreiben. Um derartige Probleme zu lösen, ist allerdings Erfahrung in der Assemblerprogrammierung und Kenntnis der Speicherbelegung des CPC nötig. Als sehr gutes Handwerkszeug haben sich zwei Bücher bewährt, die keinem »Freak« in der Sammlung fehlen sollten: das »CPC-Firmware-Handbuch« von Schneider und von Markt&Technik »ROM-Listing-CPC 464/664/6128«.

Adresse	Name	Funktion	benötigte Register
BC6B hex	CAS-NOISY	Unterdrückt Meldungen des Kassetten-Managers (»LOADING NAME BLOCK XX«) oder läßt sie wieder zu	A Wenn ungleich Null, erfolgen keine Meldungen
BC77 hex	CAS-IN-OPEN	Öffnet eine Datei zum Lesen	B Länge des Dateinamens HL Adresse des Namens DE Adresse des 2-KByte-Puffers
BC83 hex	CAS-IN-DIRECT	Lesen aus der geöffneten Datei	HL Ladeadresse
BC7A hex	CAS-IN-CLOSE	Schließen der Eingabe-Datei	Keine Übergabe
BC8C hex	CAS-OUT-OPEN	Öffnet eine Datei zum Schreiben	B Länge des Dateinamens HL Adresse des Namens DE Adresse des 2-KByte-Puffers
BC98 hex	CAS-OUT-DIRECT	Schreiben in die geöffnete Datei	HL Anfangsadresse DE Programmlänge in Byte BC Einsprungadresse (Start) A Dateityp, bei Binär-Dateien eine 2
BC8F hex	CAS-OUT-CLOSE	Schließen der Ausgabe-Datei	Keine Übergabe
Laden und Speichern in einem Block			
BCA1 hex	CAS-READ	Liest eine headerlose Datei (wirkt nur bei Kassette)	HL Ladeadresse DE Dateilänge in Byte A Satzkennzeichen
BC9E hex	CAS-WRITE	Schreibt eine headerlose Datei (wirkt nur bei Kassette)	HL Anfangsadresse DE Dateilänge in Byte A Satzkennzeichen

Tabelle. Die bei Programm-Transfer relevanten Firmware-Routinen

Aus dem ersten erfahren Sie alles Wichtige über die Nutzung von Betriebssystem-Routinen aus Maschinensprache (und zum Teil auch aus dem Basic). Das zweite Buch geht noch ein paar Schritte tiefer und zeigt Ihnen auch die trickreichsten Kniffe. Was macht man zum Beispiel, wenn ein Programm den gesamten Speicher vom Basic-Start bis hin zum DOS-RAM belegt? Wohin dann mit der Transfer-Routine? Gut, wenn man in so einem Fall mal eben nachschlagen kann, um festzustellen, daß die kleine Routine auch im Puffer des Basic-Editors bei den Adressen ACA4 bis ADA5 hex (CPC 664 und 6128: ACA8 bis AD8B hex) oder im Basic-Stack zwischen AE8B hex und B08A hex (CPC 664 und 6128: AE6F bis B06E hex) Platz findet.

RUN an den Feind

Die nächste Problemgruppe sind Programme, deren Laderoutine bereits aus Maschinencode besteht. Natürlich sind auch hier gegebenenfalls die Dateinamen für das Nachladen der restlichen Programmteile zu ändern. Mit einem Monitor-Programm ist das auch für weniger Geübte kein unüberwindliches Problem. Weit weniger erfreulich ist eine Eigenheit des Betriebssystems, die sich spätestens dann bemerkbar macht, wenn Sie das Programm mit »RUN "name"« zu laden versuchen: Nach Starten der Laderoutine ertönt im Computer ein schmatzendes »Klick« und nichts geht mehr! Was nun?

Ganz einfach, der Befehl RUN ruft beim Laden von Maschinencode-Programmen eine verhängnisvolle Firmware-Routine auf: MC-BOOT-PROGRAM. Diese wiederum lädt und startet zwar wie gewünscht das Vorprogramm, schaltet aber gleichzeitig »so ganz nebenbei« sämtliche Erweiterungs-ROMs ab. Zu jener Gattung gehört jedoch auch das DOS-ROM, so daß der Computer sofort seine diplomatischen Beziehungen zum Diskettenlaufwerk abbricht.

In den meisten Fällen hilft hier ein kurzes Basic-Programm, das den Lader ohne RUN startet und mit auf der Diskette gespeichert ist:

```
10 MEMORY ladeadresse-1
20 LOAD "name"
30 CALL startadresse
```

Als ob es nicht schon schwierig genug ist, gibt es noch subtilere Speicher-Mechanismen: Programme, bei denen das Hauptprogramm aus nur einem einzigen Block ohne Header besteht. Auch das ist ohne weiteres aus der Firmware der CPCs herauszuholen. Der Aufruf dieser Laderoutine erfolgt über die Adresse BCA1 hex, ist jedoch nur aus Maschinensprache heraus zu realisieren, da verschiedene Register benötigt werden. Zuerst müssen Sie

Fortsetzung auf Seite 162



CPC-Literatur auf einen Blick

Um seinen Computer bis ins Detail kennen und nutzen zu lernen, braucht man Literatur. Da sich nun aber fast jeder Verlag dankbar jeden neuen Themas annimmt, erscheint jeweils innerhalb kürzester Zeit eine kaum mehr überschaubare Menge neuer Bücher. So steht man dann oft hilflos vor dem Bücherregal und weiß nicht mehr aus noch ein. Um Ihnen das zu ersparen, haben wir für Sie Bücher zusammengetragen, die sich in irgendeiner Form mit dem CPC beschäftigen. Somit dürfte für jeden etwas dabei sein. Wir haben einem schnellen Überblick Rechnung getragen und die Bücher nach Fachgebieten geordnet.

(Udo Reetz)

Verlag	Titel		Preis (DM)
Einsteigerbücher			
SCD	Benutzerhandbuch CPC 464	(liegt dem 464 bei)	49,-
SCD	Benutzerhandbuch CPC 664	(liegt dem 664 bei)	69,-
SCD	Benutzerhandbuch CPC 6128	(liegt dem 6128 bei)	69,-
SCD	Benutzerhandbuch Joyce	(liegt dem Joyce bei)	69,-
DB	CPC 464 für Einsteiger		29,-
DB	CPC 6128 für Einsteiger		29,-
DB	DFÜ für jedermann		39,-
K.D.	Dumont's Handbuch zum Schneider CPC		24,80
HÜ	CPC 464 Praxis		35,-
HÜ	CPC 464/664 Praxis (Schwerpunkt Grafik)		34,-
M&T	Der Schneider CPC 6128		46,-
M&T	CPC 464 für Ein- und Umsteiger		46,-
SYB	Mein Schneider CPC (464,664)		42,-
SYB	Arbeiten mit dem Schneider CPC		38,-
VOG	Was der CPC 464 alles kann		28,-
Grafikbücher			
DB	CPC 464 Grafik und Sound		39,-
IWT	Schneider CPC Grafik		48,-
SYB	Das Schneider CPC Grafikbuch		42,-
WES	Grafik auf dem CPC 464		29,80
M&T	Schneider CPC Grafikprogrammierung		46,-
Hardware			
DB	Das Floppybuch		49,-
DB	CPC Hardware-Erweiterung		49,-
HEI	Das große Diskettenbuch zum Schneider CPC		49,-
HEI	Der perfekte Umgang mit Disketten		49,-
M&T	Hardwarebuch zum Schneider CPC 464		46,-
Tips und Tricks			
DB	CPC 464 Tips & Tricks		39,-
DB	CPC Tips & Tricks		39,-
DB	CPC Tips & Tricks Band 2		39,-
GOL	111 Tips für den Schneider-Computer		12,80
VOG	Utilities für CPC 464 und 664		28,-
VOG	Superspiele und Utilities für CPC 464 und 664		33,-
Maschinensprache			
SCD	Firmware-Handbuch		89,-
DB	CPC 464 Intern		69,-
DB	CPC 664/6128 Intern		69,-
DB	Das Maschinensprachbuch zum CPC 464		39,-
DB	Peeks & Pokes zum CPC		29,-
HUS	CPC 464 Inside Out		29,80
P.I.	CPC 464-Erweiterungen und Maschinenprogramme		32,-
M&T	CPC 464-Programmieren in Maschinensprache		46,-
M&T	ROM-Listing CPC 464/664/6128		52,-
SYB	Schneider CPC 464 Assembler-Kurs		64,-
Basic			
SCD	Basic-Handbuch	(liegt Joyce bei)	69,-
DB	Das Basic-Trainingsbuch zum CPC 464		39,-
DB	CPC 464 Basic-Programme		29,-
HEI	Das große Basic-Lexikon zum Schneider CPC 464		39,-
HEI	Das Standard-Basic-Buch zum Schneider CPC 464		49,-
HEI	Basic leicht und schnell gelernt am CPC 464		68,-
HEI	Basic-Brevier für den Schneider CPC 464		29,80
H.H.	Basic-Brevier für den CPC 464		59,-
M&T	CPC 464 Basic-Kurs		46,-
SYB	CPC 464 Mein erstes Basic-Programm		32,-

Verlag	Titel	Preis (DM)
Anwendungen		
DB	Das Schulbuch zum CPC 464	49,-
DB	Adventures und wie man sie programmiert	39,-
DB	Das Ideenbuch zum CPC 464	39,-
DB	CP/M-Trainingsbuch zum CPC 464	49,-
HEI	Mathematik mit dem Computer leichtgemacht	49,-
H.H.	Modelle der Wirklichkeit (Simulationen)	29,80
IWT	Schneider CPC Mathematik	44,-
IWT	Schneider CPC Wirtschaft	44,-
IWT	Schneider CPC dBaseII	48,-
M&T	CP/M 2.2 Anwenderhandbuch CPC 464/664/6128	46,-
M&T	Programmentwicklung unter CP/M 2.2 CPC 464/664	52,-
M&T	CP/M Plus Anwenderhandbuch CPC 664/Joyce	46,-
MVG	Praktische Programme für den Schneider CPC 464	29,80
VOG	Abenteuerspiele programmieren auf dem CPC 464	33,-
VOG	Höhere Mathematik auf dem CPC 464	33,-
VOG	Start mit LOGO auf dem CPC 464 und 664	30,-
VOG	Start in die künstliche Intelligenz	30,-
Sonderhefte		
VOG	Chip-Special Computerprogramme CPC 464 Ausgabe 1	18,-
VOG	Chip-Special Computerprogramme CPC 464 Ausgabe 2	18,-
VOG	Chip-Special Computerprogramme CPC 464 Ausgabe 3	18,-
M&T	Sonderheft 2/1985, Happy Computer	14,-
M&T	Sonderheft 1/1986, Happy Computer	14,-
M&T	Sonderheft 4/1986, Happy Computer	14,-

Hier können Sie weitere Informationen einholen

Kürzel	Verlag / Anschrift	Telefon / Ort
SCD	Schneider Computer Divison Silvastr. 1	08245/51-0 8939 Türkheim 1
DB	Data-Becker Merowingerstr. 30	0211/310010 4000 Düsseldorf 1
K.D.	K. Dumont Mittelstr. 12 - 14	0221/2053-241 5000 Köln
GOL	Goldmann Verlag Neumarkterstr. 18	089/431800 8000 München 80
HEI	Heim-Verlag Heidelberger Landstr. 194	06151/55375 6100 Darmstadt
H.H.	H. Heise-Verlag Bissendorferstr. 8	0511/5352-0 3000 Hannover 61
HUS	H. Huslik-Verlag Neidhartstr. 24	0821/571011 8900 Augsburg
HÜ	H. Hüthig-Verlag Im Weiher 10	06221/489-0 6900 Heidelberg 1
P.I.P.	Idea-Verlag Krautgartenweg 6	08105/9124 8031 Puchheim/Ort
IWT	IWT-Verlag Wendelsteinstr. 3	08106/31017 8011 Vaterstetten
M&T	Markt&Technik-Verlag Hans-Pinselstr. 2	089/4613-0 8013 Haar bei München
MVG	MVG-Verlag Justus-von-Liebigstr. 1	08191/125-1 8910 Landsberg
SIG	Signum Medien Verlag Eisenheimerstr. 52	089/5705892 8000 München 21
SYB	Sybex-Verlag Vogelsangerweg 3	0211/626441 4000 Düsseldorf 30
VOG	Vogel-Verlag Postfach 6740	0931/4102-1 8700 Würzburg 1
WES	Westermann Schulbuchverlag Postfach 3320	0531/7080 3300 Braunschweig

J.R. läßt grüßen

Nicht nur »Dallas«- und »Denver«-Süchtige werden mit »Öl« ihre wahre Freude haben.

Wie der Name schon sagt, dreht sich beim Spiel »Öl« alles um das große Geschäft mit dem flüssigen Gold. Zur Einstimmung in Ihre neue Rolle als Mitglied der großen Ewing-Familie spielt der Computer nach dem Programmstart die amerikanische Nationalhymne. Danach erhalten Sie die einmalige Chance, Ihr Glück im harten Geschäftsleben zu versuchen. Nur durch wohlüberlegte Investitionen in Ölfelder und Maschinen sowie perfekte Logistik schaffen Sie es, Ihr Startkapital zu vermehren. Da die komplette Bedienerführung menügesteuert abläuft und das Listing mit reichlich Kommentaren versehen ist, erübrigt sich eine detaillierte Bedienungsanleitung. In der Schlußbewertung erfahren Sie, ob sich eine berufliche Veränderung empfiehlt oder Sie wohl doch besser Ihren derzeitigen Job behalten sollten, um nicht den Hungertod sterben zu müssen. Damit Sie jedoch nicht verzweifeln, hat »Öl« zur Aufmunterung stets einen lockeren Spruch »auf den Lippen«.

(Andreas Gundlack/Stefan Bürgel/ja)

Programm-Steckbrief	
Name:	Öl
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
 10 REM ***********************************
    ****************                  [0C1E]
 20 REM ******* Andreas Gundlack & Stefan
    Buergel *******                   [6E8C]
 30 REM ************** 3050 Wunstorf 1 **
    ****************                  [C850]
 40 REM ******************  Oel  *******
    ****************                  [A118]
 50 REM ***********************************
    ****************                  [C826]
 60 REM *************                 [A758]
 70 REM Anfangsgrafik                 [477A]
 80 REM *************                 [295C]
 90 MODE 1:BORDER 1:INK 0,1:INK 1,26:PEN
    1:PAPER 0:CLS                     [743C]
100 SPEED KEY 20,1                    [9514]
110 PEN 3:FOR i=10 TO 17:LOCATE i,4:PRIN
    T CHR$(143):NEXT                  [BB40]
120 FOR i=4 TO 13:LOCATE 17,i:PRINT CHR$
    (143):NEXT                        [5D68]
130 FOR i=16 TO 11 STEP-1:LOCATE i,13:PR
    INT CHR$(143):NEXT                [EB38]
140 FOR i=13 TO 4 STEP-1:LOCATE 10,i:PRI
    NT CHR$(143):NEXT                 [33D2]
150 PLOT 255,176:DRAW 270,190:PLOT 269,1
    90:DRAW 255,177                   [4BA2]
160 PLOT 255,335:DRAW 270,350:PLOT 269,3
    50:DRAW 255,335                   [678E]
170 PLOT 128,335:DRAW 143,350:PLOT 144,3
    50:DRAW 128,335                   [787A]
180 PEN 2:FOR a=9 TO 16:LOCATE a,5:PRINT
     CHR$(143):NEXT                   [BBDC]
190 FOR a=5 TO 14:LOCATE 16,a:PRINT CHR$
    (143):NEXT                        [A658]
200 FOR a=16 TO 9 STEP-1:LOCATE a,14:PRI
    NT CHR$(143):NEXT                 [E1C4]
210 FOR a=14 TO 5 STEP-1:LOCATE 9,a:PRIN
    T CHR$(143):NEXT                  [3B62]
220 PEN 3:LOCATE 12,1:PRINT CHR$(143) [451A]
230 LOCATE 15,1:PRINT CHR$(143)       [A342]
240 PLOT 160,383:DRAW 175,398:PLOT 176,3
    98:DRAW 160,383                   [19B6]
250 PLOT 175,368:DRAW 190,383:PLOT 189,3
    83:DRAW 175,368                   [E0C6]
260 PLOT 175,383:DRAW 190,398:PLOT 189,3
    98:DRAW 175,383                   [04D4]
270 PLOT 209,383:DRAW 222,398:PLOT 224,3
    98:DRAW 209,383                   [90B2]
280 PLOT 223,368:DRAW 239,383:PLOT 237,3
    83:DRAW 223,368                   [7CB0]
290 PLOT 222,383:DRAW 237,398:PLOT 238,3
    98:DRAW 222,383                   [DCB8]
300 PEN 2:LOCATE 11,2:PRINT CHR$(143) [4B16]
310 LOCATE 14,2:PRINT CHR$(143)       [7D40]
320 PEN 3:FOR i=4 TO 13:LOCATE 26,i:PRIN
    T CHR$(143):NEXT                  [754C]
330 FOR i=26 TO 33:LOCATE i,13:PRINT CHR
    $(143):NEXT                       [71D2]
340 PLOT 511,176:DRAW 526,191:PLOT 525,1
    91:DRAW 511,176                   [4890]
350 PLOT 511,191:DRAW 526,206:PLOT 525,2
    06:DRAW 511,191                   [297A]
360 PLOT 399,191:DRAW 414,206:PLOT 416,2
    06:DRAW 399,191                   [0FAA]
370 PLOT 399,335:DRAW 414,350:PLOT 416,3
    50:DRAW 399,335                   [D6AC]
380 PLOT 384,335:DRAW 399,350:PLOT 400,3
    50:DRAW 384,335                   [C5A0]
390 PEN 2:FOR a=5 TO 14:LOCATE 25,a:PRIN
    T CHR$(143):NEXT                  [753A]
400 FOR a=25 TO 32:LOCATE a,14:PRINT CHR
    $(143):NEXT                       [37AC]
410 LOCATE 7,19:PRINT"Ein Spiel von AGSB
    C-Software"                       [6212]
420 LOCATE 6,22:CALL &BB9C:PRINT CHR$(16
    4);" By Gundlack & Buergel , 1985":C
    ALL &BB9C                         [71E4]
430 LOCATE 10,25:PRINT"Abbrechen mit ENT
    ER"                               [DD20]
440 GOSUB 2690                        [2152]
450 REM ************************      [C35A]
460 REM Daten fuer die Oelfelder      [E100]
470 REM ************************      [475E]
480 DATA Overquell,31760,1827,1630000 [00CC]
490 DATA Blackgold,72435,1394,1910000 [565E]
500 DATA Happyoil,28298,2437,1445000  [0BE8]
510 DATA Ewingbohr,100909,987,2040000 [3394]
520 DATA Luckyhole,49372,1987,1740000 [58C6]
530 RESTORE 480                       [9B30]
540 FOR a=1 TO 5                      [3D4E]
550 READ of$(a),ofp$(a),btf(a),gfm(a):NE
    XT a                              [C028]
560 REM *************                 [DFC2]
570 REM Hauptprogramm                 [884E]
580 REM *************                 [09C6]
590 MODE 2:PEN 1:k=150000             [C656]
600 INPUT"Wie heissen Sie ?";n$       [3856]
610 MODE 2:jj=1                       [DF54]
620 WINDOW#1,69,80,1,8:PAPER#1,1:PEN#1,0
    :CLS#1                            [14E8]
630 WINDOW#0,1,68,1,25:PAPER#0,0:PEN#0,1
    :CLS#1                            [68D2]
640 j=1985+jj                         [286E]
650 REM *****                         [5E22]
660 REM Menue                         [9D74]
670 REM *****                         [E826]
680 LOCATE 19,4:PRINT"{CTRL X}Sie haben
    die Qual der Wahl!{CTRL X}"       [5E0E]
690 LOCATE 17,7:PRINT"{CTRL X}1{CTRL X}
    Oelfeld kaufen"                   [4D86]
700 LOCATE 17,9:PRINT"{CTRL X}2{CTRL X}
    Bohrgestaenge kaufen"             [C482]
710 LOCATE 17,11:PRINT"{CTRL X}3{CTRL X}
     Pumpen kaufen"                   [AB46]
720 LOCATE 17,13:PRINT"{CTRL X}4{CTRL X}
     Tankwagen kaufen"                [F2A4]
730 LOCATE 17,15:PRINT"{CTRL X}5{CTRL X}
     Weiter"                          [0518]
740 GOSUB 2840                        [4352]
750 LOCATE 20,20:PRINT"{CTRL X} Wahl:{CT
    RL X}",:INPUT w$                  [69AC]
760 IF VAL(w$)<1 OR VAL(w$)>5 THEN 750 [45AC]
770 ON VAL(w$) GOTO 810,1000,1200,1400,1
    640                               [0340]
780 REM **************                [581E]
790 REM Oelfeld kaufen                [6F32]
800 REM **************                [6310]
810 CLS                               [9036]
820 LOCATE 26,4:PRINT"{CTRL X} Oelfeldka
    uf {CTRL X}"                      [916E]
830 xx=125:yy=370:x1=125:y1=330:y2=180:G
    OSUB 2900                         [E8A4]
840 LOCATE 19,7:PRINT"{CTRL X} Nr. {CTRL
     X}{5 SPACE}{CTRL X} Name {CTRL X}{5
     SPACE}{CTRL X} Preis {CTRL X}"   [5FDC]
850 FOR a=1 TO 5:PAPER 0:PEN 1        [8736]
860 LOCATE 20,8+a:PRINT a,of$(a),ofp$(a)
                                      [F89C]
870 NEXT a                            [9AFE]
880 LOCATE 22,18:INPUT "{CTRL X} Nr des
    Oelfeldes:{CTRL X}";nr            [FB66]
890 IF ofp$(nr)="verkauft" THEN PRINT:PR
    INT"JR was here ! Oelfeld ist futsch
    !":GOTO 960                       [AAA2]
900 k=k-VAL(ofp$(nr)):GOSUB 2840      [1ADE]
```

Listing. Haben Sie das Zeug zum Ölmagnaten?

```
910 bb=bb+1:bs$(bb)=of$(nr):bt(bb)=btf(n
    r)                                        [2BB2]
920 CLS:xx=125:yy=370:x1=125:y1=330:y2=1
    80:GOSUB 2900                             [19DC]
930 LOCATE 24,4:PRINT"{CTRL X} Sie besit
    zen nun: {CTRL X}":ofp$(nr)="verkauf
    t"                                        [64C4]
940 FOR x=1 TO bb:LOCATE 23,7+x:PRINT x;
    bs$(x):NEXT                               [7486]
950 LOCATE 23,16:PRINT"{CTRL X} Weiter m
    it Taste {CTRL X}"                        [B508]
960 CALL &BB06:GOTO 1710                      [C1CE]
970 REM ********************                  [7118]
980 REM Bohrgestaenge kaufen                  [0F3A]
990 REM ********************                  [851C]
1000 CLS                                      [1D86]
1010 FOR a=1 TO 2:p(a)=INT(RND*500)+500:
     NEXT a                                   [96FE]
1020 LOCATE 20,2:PRINT"{CTRL X} Bohrgest
     aenge kaufen {CTRL X}"                   [DB9C]
1030 LOCATE 15,4:PRINT"{CTRL X} Firma {C
     TRL X}{11 SPACE}{CTRL X} Nr {CTRL X
     }{3 SPACE}{CTRL X} Preis {CTRL X}"       [F69E]
1040 LOCATE 15,6:PRINT"Happybohr{11 SPAC
     E}1{5 SPACE}";p(1)                       [D036]
1050 LOCATE 15,7:PRINT"Tief & Schnell{6
     SPACE}2{5 SPACE}";p(2)                   [F612]
1060 PLOT 70,50:DRAWR 470,0                   [5C1E]
1070 PLOT 245,50:DRAWR 75,225:DRAWR 75,-
     225                                      [E96C]
1080 PLOT 310,245:DRAWR 0,-195:PLOT 330,
     245:DRAWR 0,-195                         [793A]
1090 PLOT 295,200:DRAWR 50,0                  [2884]
1100 PLOT 270,125:DRAWR 100,0                 [63CA]
1110 LOCATE 20,10:INPUT"{CTRL X} Firmenn
     r:{CTRL X}";nr                           [4B3C]
1120 PRINT:PRINT                              [8270]
1130 INPUT "{7 SPACE}Wieviele 500m-Stuec
     ke";st                                   [2A70]
1140 PRINT                                    [C1E6]
1150 INPUT "{7 SPACE}Auf welches Oelfeld
     ";ofn                                    [F676]
1160 IF ofn>bb THEN PRINT:PRINT"{16 SPAC
     E}Woher nehmen und nicht stehlen ?"
      ELSE 1180                               [0D8A]
1170 PRINT"{29 SPACE}{CTRL X} Taste {CTR
     L X}":CALL &BB06:GOTO 1710               [9F5A]
1180 bhg(ofn)=bhg(ofn)+st*500:k=k-st*p(n
     r):GOSUB 2840:GOTO 1710                  [2C64]
1190 REM *************                        [E722]
1200 REM Pumpen kaufen                        [C3EC]
1210 REM *************                        [9114]
1220 CLS                                      [238E]
1230 xx=125:yy=370:x1=125:y1=325:y2=230:
     GOSUB 2900                               [5AFA]
1240 FOR a=1 TO 2:p(a)=INT(RND*500)+500:
     NEXT a                                   [1C08]
1250 LOCATE 26,4:PRINT"{CTRL X} Pumpenka
     uf {CTRL X}"                             [0B3E]
1260 LOCATE 20,7:PRINT"{CTRL X} Firma {C
     TRL X}{4 SPACE}{CTRL X} Nr {CTRL X}
     {2 SPACE}{CTRL X} Preis {CTRL X}"        [34A6]
1270 LOCATE 20,9:PRINT"Everpump{4 SPACE}
     1{5 SPACE}";p(1)                         [7CCC]
1280 LOCATE 20,10:PRINT"Pump & Co{3 SPAC
     E}2{5 SPACE}";p(2)                       [DBAE]
1290 LOCATE 26,13:INPUT"{CTRL X} Firmenn
     r:{CTRL X}";nr                           [DE60]
1300 IF nr<1 OR nr>2 THEN 1290                [DF98]
1310 LOCATE 27,16:INPUT"Anzahl:";st           [BE02]
1320 LOCATE 21,18:INPUT"Auf welches Oelf
     eld";ofn                                 [BC88]
1330 IF ofn>bb THEN 1160                      [AD24]
1340 IF pu(ofn)+st>95 THEN PRINT:PRINT"M
     ehr als 95 Pumpen verkauft die Firm
     a nicht fuer EIN Oelfeld":GOTO 1310
                                              [8B08]
1350 k=k-st*p(nr):GOSUB 2840                  [9306]
1360 pu(ofn)=pu(ofn)+st:GOTO 1710             [B40E]
1370 REM ****************                     [881E]
1380 REM Tankwagen kaufen                     [FE54]
1390 REM ****************                     [CC22]
1400 CLS                                      [258E]
1410 SYMBOL AFTER 250                         [340A]
1420 SYMBOL 250,0,1,3,3,63,63,127,127         [3D42]
1430 SYMBOL 251,224,242,21,23,20,245,252
     ,255                                     [BE00]
1440 SYMBOL 252,0,168,124,220,92,92,60,2
     52                                       [3F4E]
1450 SYMBOL 253,195,153,189,60,24,255,25
     5,0                                      [C3D0]
1460 SYMBOL 254,192,152,188,60,24,255,25
     5,0                                      [53CA]
1470 SYMBOL 255,12,102,242,243,96,255,25
     5,0                                      [4CB8]
1480 FOR i=3 TO 60 STEP 3                     [6C8A]
1490 LOCATE i,22:PRINT CHR$(250);CHR$(25
     1);CHR$(252)                             [A7B4]
1500 LOCATE i,23:PRINT CHR$(253);CHR$(25
     4);CHR$(255)                             [3DB8]
1510 NEXT                                     [124C]
1520 xx=125:yy=370:x1=125:y1=325:y2=230:
     GOSUB 2900                               [5BFE]
1530 FOR a=1 TO 2:p(a)=(INT(RND*500)+500
     )*10:NEXT a                              [BBC4]
1540 LOCATE 26,4:PRINT"{CTRL X} Tankwage
     nkauf {CTRL X}"                          [AB98]
1550 LOCATE 18,7:PRINT"{CTRL X} Firma {C
     TRL X}{9 SPACE}{CTRL X} Nr {CTRL X}
     {3 SPACE}{CTRL X} Preis {CTRL X}"        [7B38]
1560 LOCATE 19,9:PRINT"Oil on Road{6 SPA
     CE}1{4 SPACE}";p(1)                      [3F06]
1570 LOCATE 19,10:PRINT"Spritdrive{7 SPA
     CE}2{4 SPACE}";p(2)                      [7666]
1580 LOCATE 24,13:INPUT"{CTRL X} Firmenn
     r:{CTRL X}";nr                           [8B60]
1590 IF nr <1 OR nr >2 THEN 1580              [1B32]
1600 LOCATE 26,16:INPUT"Anzahl:";st           [D204]
1610 LOCATE 22,18:INPUT"Auf welches Oelf
     eld";ofn:IF ofn>bb THEN 1160             [EC58]
1620 IF tw(ofn)+st>100 THEN PRINT:PRINT"
     Mehr als 100 Tankwagen sind fuer EI
     N Oelfeld nicht zugelassen!":GOTO 1
     600 ELSE k=k-st*p(nr)                    [6F80]
1630 GOSUB 2840:tw(ofn)=tw(ofn)+st:GOTO
     1710                                     [1C76]
1640 CLS                                      [219A]
1650 LOCATE 10,5:PRINT"Wohl keine Lust d
     ie Entscheidung zu treffen,was?"         [0082]
1660 LOCATE 15,6:PRINT"Na,is ja nicht me
     in Bier !"                               [BF38]
1670 LOCATE 20,10:PRINT"{CTRL X} Weiter
     mit Taste {CTRL X}":CALL &BB06           [2E62]
1680 REM **********                           [362E]
1690 REM Auswertung                           [D052]
1700 REM **********                           [7D20]
1710 FOR a=1 TO bb:CLS                        [F004]
1720 IF bbt(a)>=bt(a) THEN 1850               [CA86]
1730 LOCATE 24,2:PRINT"{CTRL X} Oelfeld:
     {CTRL X} ";bs$(a)                        [13BA]
1740 LOCATE 10,5:PRINT"{CTRL X} Bisherig
     e Bohrtiefe{3 SPACE}:";bbt(a);"m{CT
     RL X}"                                   [52AC]
1750 LOCATE 10,7:PRINT"{CTRL X} Bohrgest
     aenge noch fuer";bhg(a);"m{CTRL X}"
                                              [C28C]
1760 PLOT 50,260:DRAWR 490,0                  [1792]
1770 PLOT 400,260:DRAWR 15,45:DRAWR 15,-
     45                                       [B1FA]
1780 PLOT 405,275:DRAWR 20,0:PLOT 410,29
     0:DRAWR 10,0                             [16AC]
1790 PLOT 413,299:DRAWR 0,-39:PLOT 417,2
     99:DRAWR 0,-39                           [9CB6]
1800 PLOT 413,260:DRAWR 0,-bbt(a)/10:PLO
     T 414,260:DRAWR 0,-bbt(a)/10:PLOT 4
     15,260:DRAWR 0,-bbt(a)/10                [B450]
1810 PLOT 416,260:DRAWR 0,-bbt(a)/10:PLO
     T 417,260:DRAWR 0,-bbt(a)/10             [8CB6]
1820 IF bhg(a)>=500 THEN bhg(a)=bhg(a)-5
     00:bbt(a)=bbt(a)+500                     [3898]
1830 IF bhg(a)=0 THEN LOCATE 10,12:PRINT
     "{CTRL X} Wir brauchen Nachschub! {
     CTRL X}"                                 [990E]
1840 LOCATE 22,20:PRINT"{CTRL X} Taste {
     CTRL X}":CALL &BB06:NEXT a:gespu=0:
     gestak=0:GOTO 2220                       [37FA]
1850 CLS                                      [C4A0]
1860 LOCATE 24,2:PRINT"{CTRL X} Oelfeld:
     {CTRL X} ";bs$(a)                        [A6C2]
1870 j1=INT(RND(1)*10)+2008                   [23D4]
1880 IF j=j1 AND k>500000 THEN LOCATE 20
     ,3:PRINT"Ein Hurrikan zerstoert all
     es !":GOTO 1890 ELSE 1900                [A9E4]
1890 FOR sta=1 TO bb:pu(sta)=0:tw(sta)=0
     :NEXT sta                                [A70E]
1900 LOCATE 5,4:PRINT"{CTRL X} Pumpen{6
     SPACE}:{CTRL X}";pu(a)                   [66E6]
1910 IF bm$(a)<>"versiegt" THEN vs(a)=vs
     (a)+pu(a)*5000:bgm(a)=bgm(a)+pu(a)*
     5000                                     [61E2]
1920 LOCATE 5,6:PRINT"{CTRL X} Versandme
     nge:{CTRL X}";vs(a);" l"                 [2960]
1930 LOCATE 5,8:PRINT"{CTRL X} Tankwagen
     {3 SPACE}:{CTRL X}"tw(a)                 [0520]
1940 tk(a)=tw(a)*4890                         [59B0]
1950 pr=(INT(RND*20)+1)/10                    [7EEE]
1960 PLOT 50,260:DRAWR 490,0                  [6396]
1970 PLOT 400,260:DRAWR 15,45:DRAWR 15,-
     45                                       [3DFE]
1980 PLOT 405,275:DRAWR 20,0:PLOT 410,29
     0:DRAWR 10,0                             [DEB0]
1990 PLOT 413,299:DRAWR 0,-39:PLOT 417,2
     99:DRAWR 0,-39                           [B8BA]
2000 PLOT 413,260:DRAWR 0,-250:PLOT 414,
     260:DRAWR 0,-250:PLOT 415,260:DRAWR
      0,-250                                  [F9F0]
2010 PLOT 416,260:DRAWR 0,-250:PLOT 417,
     260:DRAWR 0,-250                         [AE1C]
2020 LOCATE 5,11:PRINT"{CTRL X} LKW-Kapa
     zitaet:{CTRL X}";tk(a);" l"              [6A28]
2030 LOCATE 54,12:PRINT"$ Preis :";pr         [B2FC]
2040 PLOT 300,11:DRAWR 200,0:PLOT 298,10
     :DRAWR 204,0:PLOT 296,9:DRAWR 208,0
                                              [3D64]
```

```
2050 PLOT 294,8:DRAWR 212,0:PLOT 292,7:D
     RAWR 216,0:PLOT 290,6:DRAWR 220,0      [39B8]
2060 PLOT 292,5:DRAWR 216,0:PLOT 294,4:D
     RAWR 212,0:PLOT 296,3:DRAWR 208,0      [AEC0]
2070 PLOT 298,2:DRAWR 204.0:PLOT 300,1:D
     RAWR 200,0                             [30CC]
2080 IF bgm(a)>=gfm(a) THEN bm$(a)="vers
     iegt"                                  [43CC]
2090 LOCATE 5,13:PRINT"{CTRL X} Bemerkun
     g:{CTRL X}";bm$(a)                     [EE3A]
2100 IF bm$(a)="versiegt" AND vs(a)>0 TH
     EN 2170                                [3CEC]
2110 IF bm$(a)<>"versiegt" THEN 2170        [0E30]
2120 LOCATE 2,21:PRINT"Wollen Sie Tankwa
     gen auf ein anderes Oelfeld stellen
      ":INPUT c$                            [3270]
2130 IF c$="n" OR c$="N" OR c$<>"j" AND
     c$<>"J" THEN CLS:GOTO 1840             [624A]
2140 INPUT"Anzahl ";f:IF f<0 OR f>tw(a)
     THEN 2140                              [29DA]
2150 INPUT"Auf welches Oelfeld ";ofn:IF
     ofn<0 OR ofn>bb OR ofn=a THEN 2150     [6D26]
2160 tw(a)=tw(a)-f:tw(ofn)=tw(ofn)+f:CLS
     :GOTO 1840                             [DDEA]
2170 LOCATE 5,16::INPUT"Wieviel l verkau
     fen ";l                                [3BC2]
2180 IF l> vs(a) OR l>tk(a) OR l<0 THEN
     PRINT"Wohl vertippt ,wah !":l=0        [3A58]
2190 vs(a)=vs(a)-l:k=k+l*pr:GOSUB 2840      [1C20]
2200 LOCATE 22,20:PRINT"{CTRL X} Taste {
     CTRL X}"                               [8654]
2210 GOTO 1840                              [7A16]
2220 FOR z=1 TO bb                          [C5F8]
2230 gespu=gespu+pu(z)                      [2E8E]
2240 gestak=gestak+tw(z)                    [9408]
2250 NEXT:CLS                               [DF88]
2260 xx=125:yy=370:x1=125:y1=325:y2=180:
     GOSUB 2900                             [B20A]
2270 LOCATE 26,4:PRINT"{CTRL X} Zusatzko
     sten {CTRL X}"                         [3456]
2280 LOCATE 17,6:PRINT"Loehne fuer";gest
     ak;"Tankwagenfahrer:"                  [50A2]
2290 LOCATE 40,7:PRINT gestak*25;"$"        [D44C]
2300 LOCATE 17,9:PRINT"Unterhaltungskost
     en fuer"                               [80DE]
2310 LOCATE 17,10:PRINT gespu;"Pumpen:{1
     3 SPACE}";gespu*15;"$"                 [2A3C]
2320 LOCATE 29,16:PRINT"{CTRL X} Taste {
     CTRL X}"                               [4272]
2330 k=k-gespu*15:k=k-gestak*25:GOSUB 28
     40:CALL &BB06                          [1828]
2340 REM ******                             [F1D2]
2350 REM Kredit                             [EDA2]
2360 REM ******                             [75D6]
2370 CLS:IF k>0 THEN 2430                   [FB90]
2380 PRINT:PRINT"{14 SPACE}Sie bekommen
     einen Kredit von der"                  [35A8]
2390 PRINT"{14 SPACE}Firma Raff, Gier &
     Co in Hoehe von 20000$"                [3E42]
2400 PRINT"{14 SPACE}Die jaehrliche Rate
      betraegt 3000$"                       [8DB2]
2410 k=k+20000:GOSUB 2840:sc=sc+20000       [29FE]
2420 REM                                    [2F98]
2430 IF sc<=0 THEN 2490                     [4CAA]
2440 LOCATE 22,9:PRINT"{CTRL X} Gesamtsc
     huld :{CTRL X}";sc                     [F96A]
2450 LOCATE 22,11::PRINT"- Rate (3000$)"
     :sc=sc-3000:k=k-3000                   [43F6]
2460 LOCATE 22,13:PRINT"{CTRL X} = neuer
      Saldo:{CTRL X}"sc                     [8864]
2470 CALL &BB06:CLS                         [C7AA]
2480 jj=jj+1:IF j=2010 THEN 2550 ELSE 6
     40                                     [21C0]
2490 k=k+(sc*(-1)):GOSUB 2840:sc=0          [BAA4]
2500 IF j=2010 THEN 2550                    [F176]
2510 jj=jj+1:GOTO 640                       [39AC]
2520 REM *************                      [C31E]
2530 REM Endauswertung                      [3BB4]
2540 REM *************                      [F922]
2550 LOCATE 30,4:PRINT"{CTRL X} Endauswe
     rtung {CTRL X}"                        [3FDC]
2560 LOCATE 30,8:PRINT"Kapital:";k          [FC7A]
2570 IF sc>0 THEN k=k-sc:LOCATE 16,10:PR
     INT"Kapital nach Abzug der Kreditsc
     hulden:";k                             [9648]
2580 LOCATE 30,12:PRINT"{CTRL X} Spieler
     :{CTRL X} ";UPPER$(n$)                 [F520]
2590 IF k<0 THEN LOCATE 22,18:PRINT"ERSC
     HIESS DICH !!!{2 SPACE}VERSAGER !!"    [994C]
2600 IF k>0 AND k<=300000 THEN LOCATE 26
     ,18:PRINT"Du bist ein PLEITEGEIER !
     "                                      [FCA0]
2610 IF k>300000 AND k<=600000 THEN LOCA
     TE 25,18:PRINT"Du bist ein GLUECKSR
     ITTER !"                               [F8FE]
2620 IF k>600000 AND k<=1000000 THEN LOC
     ATE 21,18:PRINT"Du koenntest JR Kon
     kurenz machen !"                       [653C]
2630 IF k>1000000 AND k<2000000 THEN LOC
     ATE 20,18:PRINT"Carrington haette k
     eine Chance gegen Dich"                [08CE]
2640 IF k>=2000000 THEN LOCATE 29,18:PRI
     NT"DU BIST DER CHAMP !!!":GOSUB 270
     0                                      [7074]
2650 END                                    [ED88]
2660 REM *****                              [A488]
2670 REM Hymne                              [20E8]
2680 REM *****                              [AA8C]
2690 DIM ton(28),dauer(28),pause(28)        [95E8]
2700 RESTORE 2770:FOR a=1 TO 28             [88B4]
2710 READ ton(a),dauer(a),pause(a):NEXT
     a                                      [381C]
2720 FOR a=1 TO 28                          [FB1C]
2730 a$=INKEY$:IF a$=CHR$(13) THEN RETUR
     N                                      [6404]
2740 SOUND 1,ton(a),dauer(a)                [51CA]
2750 FOR b=1 TO pause(a):NEXT:NEXT          [22D4]
2760 RETURN                                 [9E9E]
2770 DATA 319,75,1,379,50,50,478,50,50,4
     78,100,100,379,50,1,319,50,50,239,1
     00,100                                 [9CA6]
2780 DATA 190,75,1,213,50,50,239,50,50,3
     79,50,50,338,50,50,319,100,100,319,
     50,1                                   [90C2]
2790 DATA 319,50,50,190,75,75,213,25,25,
     478,50,50,253,100,100,284,50,1,253,
     50,50                                  [5834]
2800 DATA 478,50,50,478,50,50,319,50,50,
     379,50,50,478,50,50,319,50,1,379,50
     ,50                                    [0E9E]
2810 REM *************                      [3222]
2820 REM Window Inhalt                      [0DD0]
2830 REM *************                      [6C26]
2840 CLS#1                                  [0F48]
2850 PRINT#1:PRINT#1," Jahr:";j             [18BE]
2860 PRINT#1," Kapital:";k                  [A30E]
2870 PRINT#1:PRINT#1," Spieler: ":PRINT#
     1,TAB(2);UPPER$(n$)                    [1E34]
2880 RETURN                                 [B9A4]
2890 REM ************                       [45DE]
2900 REM Draw-Routine                       [CD20]
2910 REM ************                       [6CD0]
2920 PLOT xx,yy:DRAWR 260,0:DRAWR 0,-265
     :DRAWR -260,0:DRAWR 0,265              [C9EC]
2930 PLOT x1,y1:DRAWR 260,0:PLOT x1,y2:D
     RAWR 260,0                             [D402]
2940 RETURN                                 [989E]
```

Listing. Haben Sie das Zeug zum Ölmagnaten? (Schluß)

Mondsüchtig

Wer nach den Sternen greift, will dort auch gut ankommen. Bei »Lunar Lander« brauchen Sie viel Feingefühl, um Ihre Landefähre durch Felsspalten und Krater zu dirigieren.

Viel Spiel mit wenig Basic bietet Ihnen Lunar Lander. Sie übernehmen mit dem Joystick das Kommando über eine Mondfähre, die es sicher auf dafür vorgesehenen Plattformen zu landen gilt. Wenn Sie das Programm abtippen, müssen Sie aus den Listings drei Dateien erzeugen. Um Lunar Lander zu starten, verketten Sie mit »MERGE« Listing 1 und 2. Statt Listing 2 (Mondlandschaften) anzuhängen, können Sie auch die von Ihnen mit Listing 3 erzeugten Landschaften benutzen.

Nach dem Start des Programms erscheint im oberen Teil des Bildschirms das Statusboard. Sie steuern Ihre Landefähre mit drei Düsen, die mit dem Joystick zu schalten sind. Um auf den farblich gekennzeichneten Plätzen erfolgreich zu landen, müssen Sie mit Schubkraft auf eine sehr niedrige Landegeschwindigkeit drosseln. Haben Sie alle Landeplätze eines Bildes einmal angesteuert, so gelangen Sie ins nächste Bild. Immer wenn Ihre Fähre mit den landschaftlichen Tücken kollidiert oder zu hoch hinaus will, verlieren Sie Ihr Raumschiff in einer dramatischen Explosion. Das kostet Sie je nach Spielstufe mehr oder weniger Treibstoff. Bei jedem neuen Bild, das Sie erreichen, wird Ihr Treibstoffvorrat erneuert. Das Spiel ist vorbei, wenn Sie Ihren gesamten Vorrat verbrauchen.

Mit dem »Lunar Designer« stricken Sie sich Ihre Mondlandschaften selbst. Nach dem Programmstart erscheint unten links ein Pixel-Cursor, den Sie mit dem Joystick über den Bildschirm steuern. Die dabei auftretende Gerade fixieren Sie mit dem Feuerknopf. Bei waagerechten Linien müssen Sie den Feuerknopf zweimal betätigen, wenn Sie eine Plattform erzeugen wollen.

(Steffen Adomeit/Matthias Rosin)

Programm-Steckbrief	
Name:	Lunar Lander
Computer:	CPC 464
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
100 ' ***** LUNAR LANDER                     [9484]
110 ' ***** geschrieben 4/86                 [AEB6]
120 ' ***** von Steffen Adomeit              [B01A]
130 ' *****      Tel 0211/625833             [8EC0]
140 ' *****                                  [259C]
150 :                                        [DBE0]
160 MODE 1:INK 0,0:INK 1,26:INK 2,20:INK
     3,14:BORDER 0                           [8B74]
170 PEN 2:POKE &B1C8,0:PRINT TAB(5)"LUNA
    R LANDER":POKE &B1C8,1                   [0780]
180 PRINT TAB(10)"Steffen Adomeit, 4/86"
                                             [E2FC]
190 ENV 1,6,-1,20:ENV 2,15,-1,5:ENV 3,15
    ,-1,8:ENT-1,2,1,1,4,-1,1,2,1,1           [7D98]
200 DEFINT a-z:GOSUB 1160' Symbols           [B5CC]
210 o1$=CHR$(241)+CHR$(242):o2$=CHR$(243
    )+CHR$(244)' Lunarmodule                 [CD7E]
220 DIM pl(10,2):hi=1000                     [3F46]
230 :                                        [CDDE]
240 sco=0:bild=1:lev=1:c=0                   [3F48]
250 LOCATE 1,1:PRINT"{3 SPACE}HI":LOCATE
     1,2:PRINT DEC$((hi,"#####")             [B9F4]
260 RESTORE:GOSUB 990:PEN 1:LOCATE 6,5:P
    RINT"Druecke Taste fuer Spielbeginn"
                                             [0280]
270 a$=INKEY$:IF a$<>""THEN INK 1,26:LOC
    ATE 6,5:PRINT SPACE$(30):GOTO 330        [31BC]
280 c=c+1:IF c>4200 THEN c=0                 [8050]
290 INK 1,26-((c/7)MOD 26):GOTO 270          [925C]
300 INK 1,(c/7)MOD 26:GOTO 270               [3E80]
310 :                                        [DBDC]
320 GOSUB 990' Landschaft malen              [89C6]
330 GOSUB 960:TAGOFF' Score ausgeben         [2D26]
340 GOSUB 1110' Sterne malen                 [82FE]
350 tr=576:TAGOFF:LOCATE 1,3:PEN 2:PRINT
     CHR$(24)STRING$(40,240)CHR$(24):LOC
    ATE 1,3:PRINT CHR$(245)CHR$(246)CHR$
    (247)CHR$(248):PEN 1' FUEL               [02D8]
360 LOCATE 1,24:PEN 3:PRINT CHR$(24)"sco
    re"CHR$(24)                              [20F6]
370 PEN 2:LOCATE 39,1:PRINT"B";MID$(STR$
    (bild),2)                                [1D10]
380 LOCATE 39,2:PRINT"L";MID$(STR$(lev),
    2)                                       [EDA2]
390 xstep=0:ystep=0:y=348:x=2:t=0:x1=0:y
    1=0:gel=0                                [1708]
400 POKE &B338,0:PRINT CHR$(23)CHR$(1);:
    TAG                                      [A4AE]
410 :                                        [CFDE]
420 ' ***** Hauptschleife                    [9C68]
430 z=PEEK(&B4F4):IF tr=0 THEN z=0           [7A18]
440 IF gel=0 THEN t=t+1:IF t=5 THEN t=0:
    GOSUB 630                                [7E1C]
450 IF z AND 8 THEN SOUND 132,0,0,6,1,0,
    10:gel=0:GOSUB 590:IF xstep<8 THEN x
    step=xstep+2                             [CC28]
460 IF z AND 4 THEN SOUND 129,0,0,6,1,0,
    10:gel=0:GOSUB 590:IF xstep>-8 THEN
    xstep=xstep-2                            [F990]
470 IF z AND 1 THEN SOUND 130,0,0,6,1,0,
    10:gel=0:GOSUB 590:IF ystep>-6 THEN
    ystep=ystep-1                            [C47C]
480 x=x+xstep:y=y-ystep                      [BE26]
490 IF y>358 OR y<34 THEN ystep=0            [F2A2]
500 IF x>608 THEN x=608:xstep=0              [05AA]
510 IF x<0 THEN x=0:xstep=0                  [4AF0]
520 MOVE x1,y1:PRINT o1$;:MOVE x1,y1-16:
    PRINT o2$;                               [67FC]
530 POKE &B338,240:MOVE x,y:PRINT o1$;:M
    OVE x,y-16:PRINT o2$;                    [3718]
540 IF TEST(x+2,y-32)=2 AND TEST(x+28,y-
    32)=2 AND ystep<3 THEN 790               [4C22]
550 IF gel=0 THEN te=TEST(x+2,y-32)OR TE
    ST(x+28,y-32)OR TEST(x-2,y-10)OR TES
    T(x+32,y-10)OR TEST(x+4,y)OR TEST(x+
    28,y+2):IF te>1 THEN 860                 [AD38]
560 x1=x:y1=y:GOTO 430                       [95BA]
570 :                                        [DDEC]
580 ' ***** Treibstoffverbrauch              [A7A0]
590 IF tr>0 THEN tr=tr-2:MOVE tr+64,352:
    DRAWR 0,14,2                             [D7C6]
600 POKE &B338,240:RETURN                    [67CE]
610 :                                        [E3E2]
620 ' ***** Traegheit der Faehre/Mondanz
    iehung                                   [99B4]
630 IF xstep>0 THEN xstep=xstep-1            [C1F8]
640 IF xstep<0 THEN xstep=xstep+1            [CEF2]
650 IF z AND 1 THEN RETURN                   [89EE]
660 IF ystep<16 THEN ystep=ystep+1           [156A]
670 RETURN                                   [7D3A]
680 :                                        [CEF0]
690 ' ***** Alle Plattformen angeflogen      [E7B0]
700 WHILE tr>0:GOSUB 590:sco=sco+1:GOSUB
     960:SOUND 130,580-tr,0,6,1:WEND         [CE90]
710 bild=bild+1:GOTO 320                     [035E]
720 :                                        [CEE6]
730 ' ***** Spiel zuende (Kein Fuel)         [997E]
740 TAGOFF:LOCATE 14,6:PEN 1:PRINT"GAME
    OVER":SPEED INK 5,5:INK 1,0,26           [3BF4]
750 IF sco>hi THEN hi=sco:FOR i=1 TO 50:
    LOCATE 4,1:PRINT"{2 SPACE}":LOCATE 4
    ,1:PRINT"HI":NEXT                        [0B0E]
760 FOR i=1 TO 4000:NEXT:INK 1,26:LOCATE
     14,6:PRINT SPACE$(10):GOTO 240          [F0CE]
770 :                                        [CDF0]
780 ' ***** Gluecklich gelandet              [89AE]
790 SOUND 128,239,0,15,3,1:SOUND 129,190
    ,0,15,3,1:SOUND 132,159,0,15,3,1         [B2AA]
800 a=x+16:b=y-34:FOR i=0 TO pl              [0938]
810 IF pl(i,0)<>-1000 AND a>pl(i,1)AND a
    <pl(i,1)+pl(i,0)AND b<pl(i,2)AND b>p
    l(i,2)-8 THEN FOR i1=0 TO 6 STEP 2:M
    OVE pl(i,1),pl(i,2)-i1:DRAWR pl(i,0)
    -2,0,1:NEXT:sco=sco+pl(i,0):pl(i,0)=
    -1000:upl=upl-1:i=1000                   [7E22]
820 NEXT:IF upl<1 THEN 700                   [CA4C]
830 gel=1:ystep=0:xstep=0:GOSUB 960:GOTO
     550                                     [A67C]
840 :                                        [9EEC]
850 ' ***** Faehre zerstoert                 [14A6]
860 SOUND 135,0,100,15,2,0,20:i1=0:FOR i
    =1 TO 100:INK 1,26-i/4                   [FC74]
870 IF i/5=INT(i/5)THEN i1=i1 XOR 1:IF i
    1=1 THEN MOVE x,y:PRINT o1$;:MOVE x,
    y-16:PRINT o2$;ELSE MOVE x,y:PRINT C
    HR$(249)CHR$(250);:MOVE x,y-16:PRINT
     CHR$(251)CHR$(252);                     [CF00]
880 NEXT                                     [6AFE]
890 FOR i=1 TO 30+(lev-1)*20:GOSUB 590:I
    F tr>0 THEN SOUND 135,i*10,2,3 ELSE
    i=lev*30                                 [6F70]
900 NEXT:GOSUB 930:TAG:INK 1,26:FOR i=1
    TO 2000:NEXT:IF tr>20 THEN 390 ELSE
    740                                      [CE72]
910 :                                        [D7E8]
920 ' ***** Faehre loeschen                  [B460]
930 MOVE x,y:PRINT o1$;:MOVE x,y-16:PRIN
    T o2$;:TAGOFF:RETURN                     [5194]
940 :                                        [D8EE]
950 ' ***** Score ausgeben                   [A5BA]
960 TAGOFF:PEN 3:LOCATE 1,25:PRINT CHR$(
    24)DEC$((sco,"#####")CHR$(24):PEN 1:
    LOCATE 1,1:TAG:RETURN                    [433C]
970 :                                        [9DF4]
980 ' ***** Mondlandschaft malen             [869A]
990 pl=0:ORIGIN 0,0,0,640,0,350:CLG 0:OR
    IGIN 0,0,0,640,0,400:TAGOFF:PRINT CH
    R$(23)CHR$(0)                            [79B0]
1000 READ a1,b1:IF a1=-1 THEN RESTORE:le
     v=lev+1:bild=1:GOTO 1000                [F21E]
1010 p=0:READ a,b:IF a=-1 THEN upl=pl:pl
     =pl-1:RETURN                            [BE12]
1020 IF a<0 THEN a=-a:p=1:pl(pl,1)=a1:pl
     (pl,2)=b1:pl(pl,0)=ABS(a-a1):pl=pl+
     1                                       [D340]
1030 IF a<a1 THEN col=0 ELSE col=3           [1CE6]
1040 dx=a-a1:dy=b-b1:l=SQR(dx*dx+dy*dy)      [615E]
1050 FOR e=0 TO 1 STEP 2                     [3B78]
1060 MOVE a1+e*dx/l,b1+e*dy/l:DRAWR 0,-4
     00,col                                  [06A6]
1070 IF p=1 THEN MOVE(a1+e*dx/l),(b1+e*d
     y/l):DRAWR 0,-6,2                       [06D0]
1080 NEXT:a1=a:b1=b:GOTO 1010                [3EA6]
1090 :                                       [D048]
1100 ' ***** Sternenhimmel malen             [1446]
1110 POKE &B338,240:FOR i=1 TO 10:a=(RND
     *318)*2:b=(RND*175)*2                   [B46C]
1120 IF TEST(a,b)=0 THEN IF ROUND(RND)=1
      THEN MOVE a-1,b:DRAWR 3,0:MOVE a,b
     -1:DRAWR 0,3 ELSE PLOT a,b              [5E86]
1130 NEXT:RETURN                             [FC7C]
1140 :                                       [B940]
1150 ' ***** Grafikzeichen definieren        [9226]
1160 SYMBOL 240,0,0,0,102,102,102,0,0        [17FA]
```

Listing 1. Landung auf dem Mond

```
1170 SYMBOL 241,0,32,32,127,255,196,196,
     127                                 [ABC0]
1180 SYMBOL 242,30,30,16,254,255,35,35,2
     54                                  [D84E]
1190 SYMBOL 243,117,53,31,12,12,24,248,2
     48                                  [784E]
1200 SYMBOL 244,94,92,248,48,48,24,15,15
                                         [06A6]
1210 SYMBOL 245,254,254,192,192,254,192,
     192,192                             [1664]
1220 SYMBOL 246,198,198,198,198,198,198,
     254,254                             [9CB2]
1230 SYMBOL 247,254,254,192,192,254,192,
     254,254                             [E868]
1240 SYMBOL 248,192,192,192,192,192,192,
     254,254                             [EC72]
1250 SYMBOL 249,0,0,0,7,15,15,7,7        [73BC]
1260 SYMBOL 250,0,0,24,252,254,254,252,2
     52                                  [0A3A]
1270 SYMBOL 251,15,31,31,15,7,0,0,0      [2266]
1280 SYMBOL 252,252,248,248,240,224,0,0,
     0                                   [26DA]
1290 RETURN                              [BD98]
1300 :                                   [C93C]
```

Listing 1. Landung auf dem Mond (Schluß)

```
1310 ' ***** Ab hier Mondlandschaften    [C244]
1320 :                                   [C740]
1330 ' ***** bild 1                      [7218]
1340  DATA 21,90,1,100,59,88,135,60,-245
     ,60,275,68,307,80,337,100           [7D56]
1350  DATA 365,126,389,160,405,170,-479,
     170,493,180,513,186,533,184,553,170
                                         [2882]
1360  DATA 563,144,589,114,609,108,637,1
     12,-1,                              [A71A]
1370 :                                   [CE4A]
1380 ' ***** bild 2                      [DC24]
1390  DATA 21,90,1,100,19,108,55,128,85,
     160,103,184,-163,184,179,192        [728E]
1400  DATA 199,212,213,240,-279,240,293,
     224,315,202,355,188,407,174,449,168
                                         [DC68]
1410  DATA -501,168,527,164,553,130,561,
     54,561,30,-635,30,637,30,-1,        [B666]
1420 :                                   [D142]
1430 ' ***** bild 3                      [D51E]
1440  DATA 21,90,1,100,29,148,67,188,-11
     9,188,119,92,-165,92,165,234        [1AA6]
1450  DATA -219,234,243,230,271,202,279,
     154,271,104,279,50,303,16,315,8     [78AC]
1460  DATA 327,4,-393,4,411,8,431,40,439
     ,76,447,118,-495,118,501,134        [83A0]
1470  DATA 511,150,-567,150,567,168,575,
     198,599,240,625,276,637,288,-1,     [A112]
1480 :                                   [CB4E]
1490 ' ***** steps                       [9EAC]
1500  DATA 21,90,1,100,9,80,-51,80,69,80
     ,69,68,-117,68,129,68               [5BE0]
1510  DATA 129,48,-181,48,195,48,195,68,
     207,68,-255,68,255,80,273,80        [53DC]
1520  DATA -331,80,331,92,-393,92,393,10
     8,-451,108,451,124,-517,124,517,150
                                         [4730]
1530  DATA -573,150,573,186,-627,186,627
     ,210,637,210,-1,                    [25EA]
1540 :                                   [C948]
1550 ' ***** compl                       [CF7E]
1560  DATA 21,90,1,100,17,76,33,66,55,58
     ,79,52,105,56,123,74                [E97E]
1570  DATA 137,112,149,166,163,220,185,2
     56,209,278,235,290,281,298,371,304  [E51E]
1580  DATA 451,296,525,274,541,260,543,2
     36,525,214,479,212,425,218,349,238  [3730]
1590  DATA 273,232,223,202,205,160,209,1
     44,-273,144,283,144,299,132,291,108
                                         [B548]
1600  DATA 269,96,243,98,209,90,201,74,2
     07,52,221,40,-277,40,299,32         [8848]
1610  DATA 347,24,391,36,435,74,453,128,
     475,168,-571,168,595,188,615,212    [7E76]
1620  DATA 623,238,623,268,613,294,597,3
     10,603,320,623,330,637,330,-1,      [2D5E]
1630 :                                   [D648]
1640 ' ***** ????!                       [F782]
1650  DATA 21,90,1,100,19,174,-65,174,-1
     35,174,-177,174,-243,174,279,174    [8418]
1660  DATA -339,174,-383,174,-471,174,-5
     15,174,-555,174,-599,174,637,174,-1
     ,                                   [CECA]
1670 :                                   [0A50]
1680 ' ***** pferdefuss                  [B9BE]
1690  DATA 21,90,1,100,25,154,67,186,141
     ,204,-199,204,217,204,255,188       [D8F0]
1700  DATA 285,154,285,112,269,80,235,66
     ,195,66,167,54,145,32,145,10        [E6BA]
1710  DATA -287,10,307,18,329,38,361,68,
```

Listing 2. Mondlandschaften

```
     373,120,367,164,369,192,377,230     [1AF2]
1720  DATA 407,276,469,316,517,330,593,3
     22,625,330,637,330,-1,              [AC52]
1730 :                                   [D24A]
1740 ' ***** plateau                     [D322]
1750  DATA 21,90,1,100,29,108,63,140,81,
     188,89,256,-301,256,313,262         [8724]
1760  DATA 323,268,347,286,361,270,385,2
     56,-551,256,557,222,573,172,587,140
                                         [54A0]
1770 DATA -637,140,638,140,-1,           [5DF6]
1780 :                                   [0954]
1790 ' ***** ecko                        [4C98]
1800  DATA 21,90,1,100,37,150,37,50,-81,
     50,81,190,159,250,159,38            [AFE4]
1810  DATA -207,38,207,282,287,282,-339,
     282,431,282,431,134,405,134,405,258
                                         [B560]
1820  DATA 235,258,235,196,-291,196,291,
     168,235,168,235,108,-289,108,289,76
                                         [FEAA]
1830  DATA 235,76,235,28,-293,28,403,28,
     403,76,431,76,431,28,-485,28        [61BC]
1840  DATA 485,282,515,282,-559,282,575,
     282,575,256,509,256,509,204,-567,20
     4                                   [D614]
1850  DATA 567,192,509,192,509,148,-625,
     148,625,282,637,282,-1,             [4EE2]
1860 :                                   [DD52]
1870 DATA -1,                            [9368]
```

Listing 2. Mondlandschaften (Schluß)

```
100 MODE 1:INK 0,0:INK 1,26:INK 2,20:INK
     3,14:BORDER 0:PEN 1                 [C844]
110 PEN 2:POKE &B1C8,0:PRINT TAB(4)"LUNA
    R DESIGNER":POKE &B1C8,1             [67A8]
120 PRINT TAB(11)"Steffen Adomeit, 4/86"
    :LOCATE 1,25:PRINT"*****";:LOCATE 1,
    24:PRINT"*****";:PEN 1               [F984]
130 MOVE 638,0:DRAWR 0,398,1             [E07E]
140 x=1:y=100:lx=x:ly=y                  [2288]
150 DIM a(500,1):a(1,0)=x:a(1,1)=y:a(0,0
    )=x+20:a(0,1)=y-10:cur=1             [E536]
160 :                                    [CAE2]
170 ' ***** Hauptschleife                [A96C]
180 j=PEEK(&B4F4)                        [6FD8]
190 IF INKEY$="X"THEN GOSUB 290          [0BE0]
200 IF j AND 8 THEN IF x<636 THEN x=x+2
    ELSE GOSUB 300:GOTO 360              [0490]
210 IF j AND 4 AND x>0 THEN x=x-2        [170E]
220 IF j AND 2 AND y>0 THEN y=y-2        [8F12]
230 IF j AND 1 AND y<330 THEN y=y+2      [93D6]
240 MOVE lx,ly:DRAW ABS(a(cur,0)),a(cur,
    1),0                                 [AAD4]
250 MOVE x,y:DRAW ABS(a(cur,0)),a(cur,1)
    ,1                                   [4628]
260 lx=x:ly=y:GOTO 180                   [EEA4]
270 :                                    [D9E6]
280 ' ***** Knopf wurde gedrueckt        [EF2C]
290 SOUND 135,100:IF a(cur,0)=x AND a(cu
    r,1)=y THEN IF a(cur,1)=a(cur-1,1)AN
    D ABS(ABS(a(cur,0))-ABS(a(cur-1,0)))
    >36 THEN 310 ELSE RETURN             [8BA6]
300 cur=cur+1:a(cur,0)=x:a(cur,1)=y:RETU
    RN                                   [8114]
310 SOUND 135,80:a(cur,0)=-a(cur,0)      [09BC]
320 IF a(cur,0)<0 THEN FOR i=y TO y-10 S
    TEP-2:MOVE x,i:DRAW ABS(a(cur-1,0)),
    i,1:NEXT ELSE FOR i=y TO y-10 STEP-2
    :MOVE x,i:DRAW ABS(a(cur-1,0)),i,0:N
    EXT                                  [4EE2]
330 RETURN                               [A62C]
340 :                                    [CCE2]
350 ' ***** Programm erzeugen            [B976]
360 LOCATE 1,1:PRINT SPACE$(120):LOCATE
    1,1                                  [0334]
370 CALL &BB03:INPUT"name des files";a$  [3302]
380 INPUT"erste zeilennummer";z          [4A44]
390 SPEED WRITE 1:OPENOUT"LUN-"+a$:curz=
    0                                    [4F10]
400 PRINT#9,z;":":z=z+1:PRINT#9,z;"' ***
    ** "+a$:z=z+1                        [F476]
410 a$=" data ":FOR i=1 TO 8             [4290]
420 IF a(curz,0)=0 AND a(curz,1)=0 THEN
    450                                  [06FE]
430 b$=STR$(a(curz,0))+",":IF a(curz,0)>
    =0 THEN a$=a$+MID$(b$,2)ELSE a$=a$+b
    $                                    [2520]
440 b$=STR$(a(curz,1))+",":IF a(curz,1)>
    =0 THEN a$=a$+MID$(b$,2)ELSE a$=a$+b
    $                                    [6626]
450 curz=curz+1:NEXT                     [C3A6]
460 IF curz<cur THEN a$=LEFT$(a$,LEN(a$)
    -1)ELSE a$=a$+"-1,"                  [56A2]
470 PRINT#9,z;a$:z=z+1:IF curz<cur THEN
    410                                  [3B10]
480 CLOSEOUT:RUN                         [34B2]
```

Listing 3. Lunar-Designer

Feldherrn unter sich

»Feldherr« ist ein dem Brettspiel Risiko nachempfundenes Strategiespiel für zwei bis vier Personen. Es geht darum, möglichst viele der 42 Länder auf einer Weltkarte zu erobern.

Listing 1 enthält das eigentliche Spiel, von dem aus die Binärdateien RISIKOMC.OBJ und RISIKO.BIN aufgerufen werden. Listing 2 enthält den Maschinencode-Teil und erzeugt die Datei RISIKOMC.OBJ. Listing 3 schließlich beinhaltet alle Daten der einzelnen Länder und die komplette Weltkarte. Sie starten es mit »RUN 9000«.

Und nun zum Spielablauf:

Die Randfarbe, die während der Eingabe des Namens eines Spielers erscheint, macht auch im Verlauf des Spiels kenntlich, wann dieser Spieler an der Reihe ist.

Nach dieser Eingabe informiert der Computer über die Stärke der Nachschubarmeen des ersten Spielers. Danach werden Sie aufgefordert, Ihre Armeen in Ihren Ländern zu stationieren. Dies geschieht dadurch, daß Sie das Fadenkreuz auf dem Bildschirm mit den Cursortasten oder dem Joystick auf das Land bewegen, das mit Armeen besetzt werden soll. Dann drücken Sie einfach »Copy« oder »Fire«.

Armeen können allerdings nur Länder besetzen, die schon im Besitz des jeweiligen Spielers sind. Um welche es sich dabei handelt und wie viele Armeen sich in diesen Ländern bereits befinden, erfahren Sie, wenn Sie »Space« drücken.

Pro Land können Sie nicht beliebig viele Armeen stationieren. Die maximale Anzahl an Armeen, die Ihnen zugestanden wird, eröffnet Ihnen ein Druck auf die »M«-Taste.

Nun dürfen Sie aus denjenigen Ihrer Länder, die mit mehr als einer Armee besetzt sind, Ihre Mannen abziehen und anderswo einsetzen. Dies ist allerdings nur möglich, wenn eine durchgehende Verbindung zwischen den eigenen Ländern besteht.

Sie können nur Länder attackieren, die direkt an Ihr eigenes Territorium angrenzen. In dem angreifenden Land muß sich auf jeden Fall mehr als eine Armee befinden, da sonst im Falle einer Niederlage das Land unbesetzt bliebe.

Nun erscheint ein sogenanntes Würfelmenü. Dort löst der Angreifer durch Drücken von »Fire« oder »Space« den Würfelvorgang aus. Ist die Zahl der angreifenden Armeen größer oder gleich 3, so darf er dreimal würfeln, ansonsten entsprechend der Armeenzahl. Dann würfelt der Verteidiger. Die gewürfelten Zahlen werden vom Computer automatisch der Größe nach sortiert und verglichen. Für jede Zahl, die niedriger als die des Gegners ist, verliert man eine Armee. Bei zwei gleichen Zahlen verliert der Angreifer. Diese Prozedur wird nun so oft wiederholt, bis der Angreifer aufgibt oder der Verteidiger keine Armeen mehr besitzt. Im zweiten Fall muß der Angreifer mit mindestens einer Armee im eroberten Land einmarschieren.

Nun kann man weiter angreifen oder seine Armeen verschieben. Hat man sich für ein Verschieben entschlossen, ist für diese Runde kein Angriff mehr möglich, und nach dem Verschieben kommt der nächste Spieler an die Reihe.

Nach jeder Runde erhält der Spieler, der an der Reihe ist, eine bestimmte Anzahl an Armeen. Diese errechnet sich aus der Zahl der besetzten Länder geteilt durch 3, den Kontinenten, die der Spieler besitzt, und den in der letzten Runde eroberten Ländern.

Zusätzliche Armeen für Kontinente:
Amerika: 5
Europa : 3
Afrika : 6
Asien : 4
Australien : 4

Die Länder der einzelnen Kontinente sind an einer charakteristischen Färbung zu erkennen. Außerdem meldet der Computer, wenn ein Spieler einen Kontinent erobert hat. Der Computer registriert, wie viele Länder ein Spieler in einer Runde erobert. Die Anzahl der Armeen pro erobertem Land nimmt mit jeder Runde zu. Am Anfang liegt sie bei 1/10, das heißt, man muß 10 Länder erobern, um eine Zusatzarmee zu bekommen.

Je mehr Nachschub anrollt, desto höher steigt der Spieler im Rang. Nach dem Soldat folgt der Gefreite, dann der Hauptmann, General und Diktator. Es kann nun vereinbart werden, daß das Spiel zu Ende ist, sobald ein Spieler einen bestimmten Rang hat oder eine bestimmte Zeit abgelaufen ist.

(Markus Melchinger/ja)

Programm-Steckbrief	
Name:	Feldherr
Computer:	CPC 464
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
10   REM ********************************   [AE76]
20   REM **                            **   [9C5C]
30   REM **  RISIKO                    **   [9580]
40   REM **                            **   [9460]
50   REM **  produced 1984/85 by       **   [5968]
60   REM **  Markus Melchinger         **   [B846]
70   REM **  Laubachweg 24             **   [F418]
80   REM **  7958 Laupheim             **   [3E8C]
90   REM **                            **   [826A]
100  REM **  for the CPC 464           **   [5C72]
110  REM **                            **   [757C]
120  REM ********************************   [109A]
130  REM                                    [5730]
140  CLS                                    [752E]
150  DEFINT a-z                             [9B30]
160  MEMORY &8998:LOAD "!risikomc.obj":LO
     AD "!risiko.bin"                       [8670]
170  DEFINT a-z:DEF FNdeek(z)=PEEK(z)+256
     *PEEK(z+1):ds=&8D68                    [1D08]
180  REM Parametereingabe ***************   [0F2A]
190  REM                                    [A13C]
200  DATA 1,6,0,24                          [9CDA]
210  MODE 1:BORDER 0:RESTORE 200:FOR i=0
     TO 3:READ f:INK i,f:NEXT i             [A186]
220  PEN 1:PAPER 3:PRINT:PRINT" RISIKO{2
     SPACE}by Markus Melchinger{11 SPACE}
     "                                      [2FB8]
230  LOCATE 1,24:PRINT SPACE$(40);          [C76E]
240  PEN 3:PAPER 0:LOCATE 2,5:PRINT"Anzah
     l der Mitspieler (2-4): .":LOCATE 31
     ,5                                     [65F0]
250  a$=INKEY$                              [293A]
260  IF a$="" THEN 250                      [ABE6]
270  IF a$<"2" OR a$>"4" THEN SOUND 2,70,
     10:GOTO 250                            [4E9E]
280  PRINT a$                               [7ED8]
290  a=VAL(a$):DIM name$(a+1)               [F432]
300  PEN 1:PRINT:PRINT" Namen der Mitspie
     ler:":PRINT:PRINT                      [9452]
310  CALL &BB81                             [5C06]
320  FOR i=1 TO a                           [ABAE]
330   BORDER i+2                            [47F4]
340   PEN 3:PRINT i;"Soldat ";:PEN 1:PRIN
     T"...............{3 SPACE}";           [78C8]
350   LOCATE 11,8+2*i:n$=""                 [5566]
360   j=1                                   [3262]
370   a$=INKEY$:IF a$="" THEN 370           [BB70]
380   e=ASC(a$)                             [CD54]
390   IF e=&7F AND j>1 THEN j=j-1:n$=LEFT
     $(n$,LEN(n$)-1):
             PRINT CHR$(8);".";CHR$(8);:
     GOTO 370                               [2418]
400   IF e=13 THEN 440                      [9020]
410   IF e<32 OR e>126 THEN SOUND 2,70,10
     :GOTO 370                              [717C]
```

```
420  IF j<16 THEN n$=n$+a$:PRINT a$;:j=j
     +1 ELSE SOUND 2,70,10                [5D88]
430  GOTO 370                             [CF94]
440  name$(i)=n$:PRINT:PRINT              [7568]
450 NEXT i                                [3F02]
460 CALL &BB84                            [5918]
470 PRINT:PRINT" Wie lange soll gespielt
     werden"                              [FC76]
480 PRINT" 2 x <Enter> wenn kein Zeitlim
    it"                                   [7612]
490 PEN 3:PRINT" Stunden: ..";:GOSUB 338
    0:zeit=ze*60                          [9A74]
500 PRINT:PRINT" Minuten: ..";:GOSUB 338
    0:zeit=zeit+ze                        [C5BE]
510 REM Rangnamen einlesen ************   [8FB8]
520 DATA Soldat,Gefreiter,Hauptmann,Gene
    ral,Diktator                          [21A2]
530 DIM status$(5):n$="":RESTORE 520      [D2C4]
540 FOR i=1 TO 5                          [455E]
550  READ n$                              [C850]
560  status$(i)=n$+" "                    [D976]
570 NEXT i                                [2808]
580 REM Laenderverteilung **************  [DE06]
590 RANDOMIZE TIME                        [D06C]
600 FOR i=1 TO a                          [D4B0]
610  FOR j=1 TO INT(42/a)                 [3796]
620   adr=ds+11+INT(RND*42)*30            [9C70]
630   IF PEEK(adr)<>0 THEN 620            [2B0C]
640   POKE adr,i:POKE adr+1,1             [A01C]
650  NEXT j                               [5E48]
660 NEXT i                                [FB08]
670 FOR i=ds+11 TO ds+1241 STEP 30        [06B4]
680  IF PEEK(i)=0 THEN POKE i,INT(RND*a)
    +1:POKE i+1,1                         [78BC]
690 NEXT i                                [490E]
700 REM Spielbeginn ******************    [E7D2]
710 s=0:MODE 1:PEN 3:PAPER 2              [0E78]
720 DIM el(a),art$(2),art1$(2),art2$(2),
    art3$(2)                              [7268]
730 aprol=0:nl$=CHR$(13)+CHR$(10)         [249C]
740 del$=CHR$(31)+CHR$(1)+CHR$(24)+CHR$(
    14)+CHR$(2)+CHR$(18)+CHR$(10)+CHR$(1
    8)+CHR$(11)                           [0D64]
750 art$(0)=" in die ":art$(1)=" nach{3
    SPACE}":art$(2)=" in den "            [39D6]
760 art1$(0)="Die ":art1$(1)="":art1$(2)
    ="Der "                               [44F6]
770 art2$(0)=" in der ":art2$(1)=" in ":
    art2$(2)=" im "                       [8D62]
780 art3$(0)=" von der ":art3$(1)=" von
    ":art3$(2)=" vom "                    [9852]
790 IF zeit=0 THEN 810                    [0E40]
800 EVERY 3000,1 GOSUB 4220               [7916]
810 karte=&8999:saver=&8B26:darst=&8B4D:
    lva=&8B95:deleter=&8B6D:vp=&8BFA      [494A]
820 IF zeit>0 THEN EVERY 3000,1 GOSUB 42
    20                                    [A4AA]
830 CALL lva                              [3074]
840 s=s+1:aprol=aprol+1:e=FRE("0")        [1790]
850 IF s>a THEN s=1                       [089C]
860 BORDER s+2                            [50D8]
870 MODE 1:PAPER 0                        [C76E]
880 GOSUB 4110                            [FD4C]
890 IF lz=0 THEN PRINT:PRINT"Zivilist ";
    name$(s):PRINT"hat keine Laender meh
    r";                 :FOR i=1 TO 2000:NEX
    T:GOTO 840                            [9CCE]
900 el(s)=0                               [0576]
910 IF zeit>=0 THEN 1130                  [A108]
920 MODE 0:PRINT:PRINT"Zeitlimit abgelau
    fen":PRINT:PRINT:PRINT                [D578]
930 maxn=0:sa=s                           [5B36]
940 FOR s=1 TO a                          [63D2]
950  GOSUB 4110:IF n>=maxn THEN maxn=n:s
    ieger=s                               [1E02]
960 NEXT s                                [F622]
970 s=sieger:GOSUB 4110                   [541E]
980 PAPER 2:BORDER s+2:PRINT"Sieger ist"
    ;CHR$(18):PRINT status$(status);CHR$
    (18)                                  [64BA]
990 INK 4,6,24:SPEED INK 4,1:PEN 4        [9076]
1000 PRINT CHR$(18):PRINT name$(s);CHR$(
     18):PEN 1                            [023A]
1010 PAPER 0:PRINT:PRINT:PRINT:PRINT"Noc
     h ein Spiel (j/n)"                   [A906]
1020 IF INKEY(45)=-1 THEN 1070            [173C]
1030 FOR i=ds+11 TO ds+1241 STEP 30       [5902]
1040  POKE i,0:POKE i+1,0                 [8386]
1050 NEXT i                               [CB5C]
1060 RUN 170                              [9D28]
1070 IF INKEY(46)=-1 THEN 1020            [7B3E]
1080 PRINT:PRINT"Altes Spiel":PRINT"fort
     setzen{5 SPACE}(j/n)"                [F8DC]
1090 IF INKEY(46)=-1 THEN 1110            [9C42]
1100 IF INKEY$<>"" THEN 1100 ELSE PRINT:
       END                                [5B96]
1110 IF INKEY(45)=-1 THEN 1090            [F840]
1120 s=sa:GOSUB 4110:BORDER s+2:zeit=0:M
     ODE 1                                [7F06]
1130 PEN 1:PRINT:PRINT " ";status$(statu
     s);                                  [6FBC]
1140 PEN 3:PRINT name$(s):PEN 1:PRINT:PR
     INT                                  [7710]
1150 PRINT" Fuer besetzte Laender{8 SPAC
     E}:";                                [6080]
1160 PEN 3:PRINT nl                       [B4BE]
1170 PEN 1:PRINT                          [43C8]
1180 PRINT" Fuer vorher eroberte Laender
      :";                                 [5916]
1190 PEN 3:PRINT nel:PEN 1:PRINT          [5EF8]
1200 PRINT" Fuer Kontinente{14 SPACE}:";
                                          [DCB4]
1210 PEN 3:PRINT nk:PEN 1:PRINT:PRINT     [59AC]
1220 MOVE 16,240:DRAWR 512,0,3            [2724]
1230 PRINT" Summe{24 SPACE}:";            [A8A6]
1240 PEN 3:PRINT n                        [4EE4]
1250 IF zeit=0 THEN 1270                  [92F2]
1260 PRINT:PRINT:PRINT:PRINT" Noch zu sp
     ielende Zeit:";zeit;"Min."           [B4BA]
1270 m$=" ":GOSUB 4060                    [D580]
1280 IF n=0 THEN 1400                     [0152]
1290 m$="Bitte verteilen Sie Ihre"+STR$(
     n)+" Armeen"                         [3376]
1300 auswahl=1:GOSUB 2770                 [BEE2]
1310 IF b=s THEN 1340                     [C6BA]
1320 m$=art1$(art)+ln$+" gehoert ihnen n
     icht !"+nl$+"Bitte neu auswaehlen"   [CCF8]
1330 GOTO 1300                            [B508]
1340 PRINT del$;"Wieviel Armeen";art$(ar
     t);ln$                               [C262]
1350 PRINT "(alt:";az;"max.";maxima;"Arm
     .:";n;") ..";                        [D39A]
1360 GOSUB 3380                           [60B0]
1370 IF ze<0 OR ze>n OR az+ze>maxima THE
     N 1340                               [E612]
1380 n=n-ze:POKE ca-15,az+ze              [3004]
1390 GOTO 1280                            [A322]
1400 m$="Wollen Sie noch verschieben (j/
     n)"                                  [052E]
1410 auswahl=0:GOSUB 2770                 [7AE4]
1420 IF a$="j" THEN GOSUB 3770:GOTO 1400
                                          [AD7C]
1430 IF a$<>"n" THEN 1410                 [FEFA]
1440 REM Angriff ***                      [F390]
1450 m$="Wollen Sie jemand"+nl$+"angreif
     en (j/n)"                            [38F8]
1460 auswahl=0:GOSUB 2770                 [8AEE]
1470 IF a$<>"n" THEN 1530                 [0B08]
1480 m$="Wollen Sie noch verschieben (j/
     n)"                                  [E53E]
1490 auswahl=0:GOSUB 2770                 [CAF4]
1500 IF a$="j" THEN GOSUB 3770:GOTO 1480
                                          [258A]
1510 IF a$<>"n" THEN 1490                 [F808]
1520 GOTO 830                             [6BB8]
1530 IF a$<>"j" THEN 1450                 [23FC]
1540 m$="Welches Land wollen Sie angreif
     en ?"                                [DE0C]
1550 auswahl=1:GOSUB 2770                 [7AF0]
1560 IF b<>s THEN 1600                    [9A40]
1570 m$=art1$(art)+ln$+" gehoert Ihnen s
     chon"+nl$                            [3716]
1580 GOSUB 4060                           [C9B0]
1590 GOTO 1450                            [7E24]
1600 ba=b:aza=az:maximaa=maxima:lca=lc:l
     na$=ln$:caa=ca:arta=art              [9E72]
1610 m$="Von welchem Land wollen Sie ang
     reifen"                              [3D24]
1620 auswahl=1:GOSUB 2770                 [BDEC]
1630 IF az<2 THEN m$="Nur 1 Armee"+art2$
     (art)+ln$+nl$:GOSUB 4060:GOTO 1450   [1E4C]
1640 IF b=s THEN 1680                     [F7D4]
1650 m$=art1$(art)+ln$+" gehoert Ihnen n
     icht"+nl$                            [2E0A]
1660 GOSUB 4060                           [D6AE]
1670 GOTO 1450                            [2C22]
1680 FOR i=ca-9 TO ca-1                   [7196]
1690  IF lca=PEEK(i) THEN 1740            [998C]
1700 NEXT i                               [D060]
1710 m$="Keine Verbindung zwischen "+lna
     $+nl$+"und "+ln$+" "                 [E90E]
1720 GOSUB 4060                           [7BA8]
1730 GOTO 1450                            [781C]
1740 REM Kampf ***                        [81FA]
1750 MODE 1:PAPER 0                       [14CA]
1760 e=status:s=ba:GOSUB 4110:s=b:status
     a=status:status=e                    [AFDA]
1770 PRINT status$(status);name$(b);:PEN
      1:PRINT " greift";art3$(art)        [2DE4]
1780 PRINT ln$;" aus mit";az-1;"Armee";:
     IF az>2 THEN PRINT "n" ELSE PRINT    [9EFA]
1790 PRINT lna$;" an, das von"            [6AC4]
1800 PEN 3:PRINT status$(statusa);name$(
     ba);:PEN 1:PRINT" mit";aza;"Armee";
                                          [80E2]
1810 IF aza>1 THEN PRINT "n" ELSE PRINT   [F210]
1820 IF az-aza>8 THEN PRINT"heldenhaft "
     ;                                    [B904]
1830 PRINT "verteidigt wird."             [3A30]
1840 a$=""                                [D2E6]
1850 FOR i=1 TO 6:a$=a$+MID$(STR$(i),2,1
     ):NEXT i                             [91E8]
```

Listing 1. Erobern Sie die Welt

```
1860 LOCATE 16,8:PEN 2:PRINT a$;"{2 SPAC
     E}";a$;"{2 SPACE}";a$                [7BD6]
1870 PEN 1:PRINT:PRINT "Angriff{6 SPACE}
     :"                                   [D09A]
1880 PRINT:PRINT "Verteidigung :"         [EDE0]
1890 PEN 3:w=1:LOCATE 16,10:FOR i=1 TO 6
     :w(i)=0:NEXT i                       [E1B0]
1900 PRINT CHR$(231);:w(w)=1:EI:i=1       [77F4]
1910 MOVE 0,304:DRAWR 640,0:DRAWR -6,-4:
     DRAWR -628,0:DRAWR -6,4              [2ADE]
1920 MOVE 0,186:DRAWR 640,0:DRAWR -6,4:D
     RAWR -628,0:DRAWR -6,-4              [FAF0]
1930 IF INKEY(47)=-1 AND JOY(0)<>16 THEN
     1930                                 [422C]
1940 EVERY 3,0 GOSUB 2020                 [F958]
1950 IF INKEY(47)<>-1  OR JOY(0)=16 THEN
     1950                                 [4010]
1960 e=REMAIN(0):e=0:i=4                  [7CDE]
1970 AFTER i,0 GOSUB 2020                 [BF98]
1980 IF e=0 THEN 1980                     [6E68]
1990 e=0:i=i+INT(RND*3)                   [156C]
2000 IF i>15 THEN 2070                    [B2AC]
2010 GOTO 1970                            [8D1A]
2020 e=1:w(w)=w(w)+1:SOUND 2,50,2         [5D18]
2030 IF w(w)<>7 THEN PRINT CHR$(231);:FO
     R j=1 TO 5:NEXT j:GOTO 2060          [28A4]
2040 FOR j=1 TO 5:PRINT CHR$(8);CHR$(16)
     ;:NEXT j                             [7900]
2050 w(w)=1                               [FC28]
2060 RETURN                               [8D90]
2070 IF w>3 THEN 2180                     [B974]
2080 IF w=1 AND az>2 THEN LOCATE 24,10:w
     =2:GOTO 1900                         [DA26]
2090 IF w=2 AND az>3 THEN LOCATE 32,10:w
     =3:GOTO 1900                         [042C]
2100 IF az=2 THEN GOTO 2150               [DCD8]
2110 IF az=3 THEN IF w(1)<w(2) THEN e=w(
     1):w(1)=w(2):w(2)=e:GOTO 2150
                        ELSE 2150         [76C2]
2120 IF w(3)>w(2) THEN e=w(2):w(2)=w(3):
     w(3)=e                               [B42A]
2130 IF w(2)>w(1) THEN e=w(1):w(1)=w(2):
     w(2)=e                               [9A20]
2140 IF w(3)>w(2) THEN e=w(2):w(2)=w(3):
     w(3)=e                               [7A2E]
2150 LOCATE 16,10                         [5D68]
2160 FOR i=1 TO 3:PRINT STRING$(w(i),CHR
     $(231));SPACE$(8-w(i));:NEXT i       [7A88]
2170 w=4:LOCATE 16,12:GOTO 1900           [8D6E]
2180 IF w=4 AND aza>1 THEN LOCATE 24,12:
     w=5:GOTO 1900                        [8BF8]
2190 IF w=5 AND aza>2 THEN LOCATE 32,12:
     w=6:GOTO 1900                        [0FFE]
2200 IF aza=1 THEN GOTO 2250              [C59C]
2210 IF aza=2 THEN IF w(4)<w(5) THEN e=w
     (4):w(4)=w(5):w(5)=e:GOTO 2250
                        ELSE 2250         [CCAC]
2220 IF w(6)>w(5) THEN e=w(5):w(5)=w(6):
     w(6)=e                               [BF50]
2230 IF w(5)>w(4) THEN e=w(4):w(4)=w(5):
     w(5)=e                               [9F46]
2240 IF w(6)>w(5) THEN e=w(5):w(5)=w(6):
     w(6)=e                               [A954]
2250 LOCATE 16,12                         [766E]
2260 FOR i=4 TO 6:PRINT STRING$(w(i),CHR
     $(231));SPACE$(8-w(i));:NEXT i       [8396]
2270 av=0:ava=0                           [7D1C]
2280 FOR i=1 TO 3                         [36C0]
2290  IF w(i)=0 OR w(i+3)=0 THEN 2320     [40DA]
2300  IF w(i)>w(i+3) THEN ava=ava+1 ELSE
      av=av+1                             [FCB4]
2310 NEXT i                               [BC5C]
2320 LOCATE 1,16:PEN 3                    [B7E6]
2330 PRINT status$(status);name$(b);" ve
     rliert"                              [2722]
2340 PEN 1:PRINT av;:PEN 3:PRINT "von";a
     z-1;"Armeen"                         [229E]
2350 PRINT status$(statusa);name$(ba);"
     verliert"                            [68AA]
2360 PEN 1:PRINT ava;:PEN 3:PRINT "von";
     aza;"Armeen"                         [5C6A]
2370 az=az-av:aza=aza-ava:POKE ca-15,az:
     POKE caa-15,aza                      [808E]
2380 PRINT:PRINT                          [5F82]
2390 IF az>1 THEN 2430                    [FB3E]
2400 PRINT "Angriff gescheitert.":m$=""   [718A]
2410 GOSUB 4060                           [3EA2]
2420 m$="Wollen Sie trotzdem nochmal"+nl
     $+"angreifen (j/n)":GOTO 1460        [7DC8]
2430 IF aza=0 THEN m$="":GOSUB 4060:GOTO
      2500                                [1A80]
2440 PRINT "Noch ein Versuch ? (j/n)"     [5B2C]
2450 a$=INKEY$:IF a$<>"" THEN 2450        [5B6E]
2460 a$=INKEY$:IF a$="" THEN 2460         [8DF8]
2470 IF a$="j" THEN 1750                  [6C90]
2480 IF a$<>"n" THEN 2450                 [DA10]
2490 m$="Noch ein Angriff (j/n)":GOTO 14
     60                                   [C030]
2500 MODE 1:PRINT                         [5348]
2510 PRINT"GRATULIERE":PRINT:PRINT        [9576]
2520 PRINT"Sie haben hiermit ";status$(s
     tatusa);name$(ba)                    [2D30]
2530 PRINT"aus ";lna$;" vertrieben.":PRI
     NT:PEN 1:PRINT "Es lebe ";
            status$(status);name$(b):
     PEN 3:PRINT:PRINT:PRINT              [F002]
2540 IF statusa>status THEN PRINT"Nieder
      mit ";status$(statusa);name$(ba):P
     RINT:PRINT                           [443A]
2550 POKE caa-16,b:el(s)=el(s)+1          [33FE]
2560 a$="":CALL lva                       [1758]
2570 IF PEEK(ds+1284)=b AND PEEK(ds+1285
     )<>b THEN a$="Amerika"               [59D2]
2580 IF PEEK(ds+1287)=b AND PEEK(ds+1288
     )<>b THEN a$="Europa"                [B844]
2590 IF PEEK(ds+1290)=b AND PEEK(ds+1291
     )<>b THEN a$="Afrika"                [DCF2]
2600 IF PEEK(ds+1293)=b AND PEEK(ds+1294
     )<>b THEN a$="Asien/Australien"      [26C0]
2610 IF a$="" THEN 2630                   [E9B0]
2620 PRINT a$;" gehoert nun Ihnen":PRINT
     :PRINT                               [9C74]
2630 a$=""                                [ACE2]
2640 IF PEEK(ds+1284)=0 AND PEEK(ds+1285
     )=ba THEN a$="Amerika"               [B8B2]
2650 IF PEEK(ds+1287)=0 AND PEEK(ds+1288
     )=ba THEN a$="Europa"                [0524]
2660 IF PEEK(ds+1290)=0 AND PEEK(ds+1291
     )=ba THEN a$="Afrika"                [89D2]
2670 IF PEEK(ds+1293)=0 AND PEEK(ds+1294
     )=ba THEN a$="Asien/Australien"      [6CB2]
2680 IF a$="" THEN 2700                   [C2BA]
2690 PRINT a$;" gehoert nun nicht mehr":
     PRINT status$(statusa);name$(ba)     [9E78]
2700 PRINT del$;"Mit wieviel Armeen woll
     en Sie";art$(arta)                   [EF6C]
2710 PRINT lna$;" einmarschieren (max.";
     STR$(MIN(az-1,maximaa));")..";       [2AFC]
2720 GOSUB 3380                           [0CB2]
2730 IF ze>MIN(az-1,maximaa) OR ze<1 THE
     N 2700                               [4F2A]
2740 POKE ca-15,az-ze                     [944C]
2750 POKE caa-15,ze                       [5F00]
2760 m$="Noch ein Angriff (j/n)":GOTO 14
     60                                   [8D30]
2770 REM Upro Search (Tastaturauswertung
      und evtl. Laendererkennung)         [758C]
2780 REM                                  [4BAA]
2790 DRAWR 0,0,3                          [21FA]
2800 CALL karte:CALL saver:CALL darst     [6AD0]
2810 PRINT del$;m$;                       [33F0]
2820 IF INKEY(0)=-1 AND JOY(0)<>1 THEN 2
     870                                  [8A50]
2830 y=PEEK(ds+1296)+2 'oben              [2F6A]
2840 IF y>200 THEN y=200                  [1B3A]
2850 CALL deleter:POKE ds+1296,y:CALL sa
     ver:CALL &BD19:CALL darst            [DA92]
2860 GOTO 2820                            [A22A]
2870 IF INKEY(2)=-1 AND JOY(0)<>2 THEN 2
     920                                  [4F58]
2880 y=PEEK(ds+1296)-2 'unten             [3684]
2890 IF y<23 THEN y=23                    [618C]
2900 CALL deleter:POKE ds+1296,y:CALL sa
     ver:CALL &BD19:CALL darst            [978A]
2910 GOTO 2820                            [4722]
2920 IF INKEY(1)=-1 AND JOY(0)<>8 THEN 3
     020                                  [EB4A]
2930 CALL deleter                         [AE1E]
2940 x=FNdeek(ds+1272)+4 'rechts          [173E]
2950 IF x<313 THEN 2990                   [EC5A]
2960 x=x-60:kp=FNdeek(ds+1299)+60         [00D2]
2970 IF kp>632 THEN kp=kp-632             [43FE]
2980 POKE &927B,kp MOD 256:POKE &927C,IN
     T(kp/256):CALL karte                 [6080]
2990 POKE ds+1272,x MOD 256:POKE ds+1273
     ,INT(x/256)                          [90EC]
3000 CALL saver:CALL darst                [C3A8]
3010 GOTO 2820                            [3A12]
3020 IF INKEY(8)=-1 AND JOY(0)<>4 THEN 3
     120                                  [2042]
3030 CALL deleter                         [1F0E]
3040 x=FNdeek(ds+1272)-4 'links           [8362]
3050 IF x>0 THEN 3090                     [4170]
3060 x=x+60:kp=FNdeek(ds+1299)-60         [6DC2]
3070 IF kp<0 THEN kp=kp+632               [6910]
3080 POKE &927B,kp MOD 256:POKE &927C,IN
     T(kp/256):CALL karte                 [D770]
3090 POKE ds+1272,x MOD 256:POKE ds+1273
     ,INT(x/256)                          [95DC]
3100 CALL saver:CALL darst                [33AA]
3110 GOTO 2820                            [9214]
3120 IF INKEY(47)<>-1 THEN maxi=0:GOSUB
     3490:GOTO 2800                       [A776]
3130 IF INKEY(38)<>-1 THEN maxi=-1:GOSUB
      3490:GOTO 2800                      [99D4]
3140 IF (INKEY(9)<>-1 OR JOY(0)=16) AND
     auswahl THEN 3190                    [D60C]
3150 IF auswahl THEN 2820                 [3490]
3160 IF INKEY(45)<>-1 THEN a$="j":RETURN
                                          [EB48]
3170 IF INKEY(46)<>-1 THEN a$="n":RETURN
                                          [0C54]
```




```
3180 GOTO 2820                                  [9722]
3190 SOUND 1,150,5:SOUND 1,100,7:SOUND 1
     ,50,5                                      [5048]
3200 CALL deleter                               [ED0C]
3210 lc=PEEK(&8D56)                             [FDEA]
3220 IF lc<>15 AND lc<>255 AND lc<>240 T
     HEN 3240                                   [1BCE]
3230 lc=PEEK(&8D58)                             [F3F2]
3240 IF INKEY$<>"" THEN 3240                    [7864]
3250 IF lc=0 THEN m1$=m$:m$="Wasser !!"+
     nl$:GOSUB 4060:m$=m1$:CALL darst:
            GOTO 2810                           [A300]
3260 FOR ca=ds+27 TO ds+27+41*30 STEP 30
                                                [8850]
3270  IF PEEK(ca)=lc THEN 3320                  [9270]
3280 NEXT ca                                    [7E20]
3290 PRINT del$;"Keine Identifikation mo
     eglich"                                    [3B32]
3300 PRINT"Bitte nochmals auswaehlen";          [4E58]
3310 CALL darst:GOTO 2820                       [9E40]
3320 ln$=""                                     [D7CE]
3330 FOR j=ca+1 TO ca+12                        [9ED8]
3340  ln$=ln$+CHR$(PEEK(j))                     [6840]
3350 NEXT j                                     [DC68]
3360 b=PEEK(ca-16):az=PEEK(ca-15):maxima
     =PEEK(ca-14):art=PEEK(ca+13)               [417E]
3370 RETURN                                     [AA9A]
3380 REM Upro Input ****                        [8220]
3390 PRINT CHR$(8);CHR$(8);:PEN 1:CALL &
     BB81                                       [2B3A]
3400 i=0:e(1)=0:e(2)=0                          [6EB4]
3410 a$=INKEY$:IF a$="" THEN 3410               [2BE8]
3420 IF INKEY(79)<>-1 AND i<>0 THEN e(i)
     =0:i=i-1:PRINT CHR$(8);".";CHR$(8);
     :          GOTO 3410                       [2B24]
3430 IF INKEY(18)<>-1 OR JOY(0)=16 THEN
     3470                                       [34C0]
3440 IF a$<"0" OR a$>"9" OR i=2 THEN 341
     0                                          [C2C0]
3450 PRINT a$;:i=i+1:e(i)=VAL(a$)               [AF9A]
3460 GOTO 3410                                  [771C]
3470 IF i=1 THEN ze=e(1) ELSE ze=e(1)*10
     +e(2)                                      [44E0]
3480 PEN 3:CALL &BB84:RETURN                    [7996]
3490 REM Upro Armeenzahl darstellen ****
     ******************                         [4B96]
3500 CALL deleter                               [2212]
3510 IF maxi THEN m1$="Maximale Armeenza
     hl" ELSE m1$="Ihre Armeen"                 [A0F4]
3520 FOR j=1 TO 3                               [3ABE]
3530 PRINT del$;m1$                             [07DC]
3540  TAG                                       [A8D0]
3550  FOR k=ds+11 TO ds+1241 STEP 30            [B858]
3560 IF NOT(maxi) AND PEEK(k)<>s THEN 36
     40                                         [8656]
3570   x=FNdeek(k+3)-FNdeek(ds+1299)            [125A]
3580   IF x<-320 THEN x=x+640                   [606C]
3590   IF x<0 OR x>312 THEN 3640                [AA5A]
3600   y=PEEK(k+5):TAG                          [4D6C]
3610   MOVE 2*x+8,2*y                           [1272]
3620   IF maxi THEN DRAWR 0,0,1:a$=STR$(
     PEEK(k+2)):PRINT RIGHT$(a$,LEN(a$)-
     1);:          GOTO 3640                    [DAC4]
3630   DRAWR 0,0,3:a$=STR$(PEEK(k+1)):PR
     INT RIGHT$(a$,LEN(a$)-1);                  [7F6E]
3640  NEXT k                                    [07AE]
3650  IF INKEY$<>"" THEN 3650                   [22B8]
3660  TAGOFF:PRINT "Bitte Taste druecken
     ";                                         [E56C]
3670  WHILE INKEY$="":WEND                      [426C]
3680  x=FNdeek(ds+1299)                         [6D36]
3690  x=x+212                                   [F5FE]
3700  IF x>632 THEN x=x-632                     [B1DC]
3710  POKE ds+1299,x MOD 256                    [7624]
3720  POKE ds+1300,INT(x/256)                   [009A]
3730  CALL karte                                [F7C0]
3740 NEXT j                                     [A26E]
3750 PEN 3:DRAWR 0,0,3                          [4DD4]
3760 RETURN                                     [8CA0]
3770 REM Upro Armeen verschieben ******         [BFBA]
3780 m$="Von welchem Land wollen Sie Arm
     een"+nl$+"abziehen"                        [2C46]
3790 auswahl=1:GOSUB 2770                       [6500]
3800 IF b=s THEN 3830                           [F0D2]
3810 IF v>5 THEN a$=" (noch)" ELSE a$=""
                                                [0F02]
3820 m$=art1$(art)+ln$+" gehoert Ihnen"+
     a$+nl$+"nicht. ":GOSUB 4060:RETURN         [8C22]
3830 IF az=1 THEN m$="Nur eine Armee"+ar
     t2$(art)+ln$+nl$:GOSUB 4060:RETURN         [4BAE]
3840 v=az-1                                     [FFB6]
3850 PRINT del$;"Wieviele Armeen wollen
     Sie";art3$(art);nl$;ln$;" abziehen
     (max.";v;") ..";:GOSUB 3380                [5FAA]
3860 IF ze>v THEN 3850                          [3BE4]
3870 IF ze=0 THEN RETURN                        [CA78]
3880 v=ze:POKE ca-15,az-v:cav=ca                [6208]
3890 m$="Sie haben"+STR$(v)+" Armeen"+nl
     $+"In welches Land soll verschoben
     werden"                                    [5504]
3900 auswahl=1:GOSUB 2770                       [B8F2]
3910 IF b=s THEN 3940                           [F2DA]
3920 IF v>5 THEN a$=" (noch)" ELSE a$=""
                                                [EA06]
3930 m$=art1$(art)+ln$+" gehoert Ihnen"+
     a$+nl$+"nicht. ":GOSUB 4060:GOTO 38
     90                                         [8FC0]
3940 IF ca=cav THEN 4000                        [FB1A]
3950 CALL vp,s,cav,lc                           [AB26]
3960 IF PEEK(ds+1271)=255 THEN 4000             [A1E0]
3970 m$="Kein Verschieben moeglich"+nl$         [6056]
3980 GOSUB 4060                                 [4DBC]
3990 GOTO 3890                                  [8644]
4000 vmax=MIN(v,maxima-az)                      [7F72]
4010 PRINT del$;"Wieviel Armeen";art$(ar
     t);ln$;nl$;"(max.";vmax;") ..";            [7DB0]
4020 GOSUB 3380                                 [69A8]
4030 IF ze>vmax THEN 4010                       [8246]
4040 POKE ca-15,az+ze:v=v-ze                    [FF1C]
4050 IF v>0 THEN 3890 ELSE RETURN               [5910]
4060 REM Upro Warten auf Tastendruck ***
     *                                          [9AE2]
4070 PRINT del$;m$;"Bitte Taste druecken
     ";                                         [2E02]
4080 IF INKEY$<>"" THEN 4080                    [1A70]
4090 WHILE INKEY$="":WEND                       [FD26]
4100 RETURN                                     [808A]
4110 REM Upro Nachschub berechnen ***           [3F4E]
4120 lz=PEEK(ds+1282-s)                         [ABDE]
4130 nl=INT(lz/3):IF nl<4 THEN nl=3             [DEDE]
4140 nk=0                                       [895E]
4150 FOR i=ds+1284 TO ds+1293 STEP 3            [C296]
4160  IF PEEK(i)=s THEN nk=nk+PEEK(i-1)         [C51E]
4170 NEXT i                                     [A668]
4180 nel=INT(el(s)*aprol/10)                    [6A24]
4190 n=nl+nk+nel                                [E2C2]
4200 IF n*a<75 THEN status=INT(n*a/15)+1
      ELSE status=5                             [7856]
4210 RETURN                                     [8D8E]
4220 REM Upro Zeit ***                          [C6D8]
4230 SOUND 1,15,3:zeit=zeit-1:RETURN            [CD36]
```

Listing 1. Erobern Sie die Welt (Schluß)

```
10 DATA 243,237,115,113,141,49,129,1027    [0072]
20 DATA 146,237,75,123,146,221,33,981      [54C2]
30 DATA 99,141,17,80,192,237,83,849        [EE22]
40 DATA 100,141,17,160,192,237,83,930      [46AE]
50 DATA 102,141,151,221,119,11,221,966     [4EFE]
60 DATA 119,5,33,64,1,151,237,610          [E31E]
70 DATA 66,33,1,0,48,68,33,249             [2A0E]
80 DATA 128,2,237,66,151,203,60,847        [6600]
90 DATA 203,29,203,31,203,60,203,932       [BE3C]
100 DATA 29,203,31,203,63,203,63,795       [A648]
110 DATA 203,63,203,63,203,63,203,1001     [08E2]
120 DATA 63,35,34,111,141,221,119,724      [079E]
130 DATA 6,42,100,141,17,80,0,386          [420C]
140 DATA 151,237,82,237,91,111,141,1050    [946A]
150 DATA 25,221,203,5,134,209,221,1018     [D4F8]
160 DATA 126,6,221,114,7,214,4,692         [DD84]
170 DATA 131,71,24,26,209,122,22,605       [4840]
180 DATA 0,25,43,237,66,9,35,415           [1FD0]
190 DATA 56,244,87,151,237,66,125,966      [2DF0]
200 DATA 1,80,0,42,100,141,237,601         [DA52]
210 DATA 66,71,62,3,184,210,196,792        [B504]
220 DATA 138,114,35,5,5,5,5,307            [AB54]
230 DATA 120,237,75,100,141,183,237,1093
                                           [40D4]
240 DATA 66,9,71,56,232,120,50,604         [8F8A]
250 DATA 109,141,237,75,123,146,33,864     [092C]
260 DATA 64,1,151,237,66,48,21,588         [51A8]
270 DATA 42,109,141,17,64,1,9,383          [202E]
280 DATA 237,82,221,126,6,254,0,926        [FBFC]
290 DATA 194,13,138,43,195,13,138,734      [31D6]
300 DATA 42,100,141,17,0,8,25,333          [29FC]
310 DATA 34,100,141,48,13,42,102,480       [6C2A]
320 DATA 141,34,100,141,17,80,0,513        [C8C0]
330 DATA 25,34,102,141,42,102,141,587      [179C]
340 DATA 17,208,199,151,237,82,32,926      [CEDA]
350 DATA 6,237,123,113,141,251,201,1072    [8D5C]
360 DATA 221,203,5,70,194,238,137,1068     [F024]
370 DATA 58,109,141,111,38,0,237,694       [C764]
380 DATA 75,123,146,17,64,1,9,435          [D53C]
390 DATA 25,43,62,2,188,56,222,598         [90B4]
400 DATA 32,18,62,128,189,56,215,700       [895E]
410 DATA 32,11,237,75,123,146,51,675       [AF56]
420 DATA 33,1,0,195,13,138,209,589         [7596]
430 DATA 22,0,25,24,225,122,209,627        [25DC]
440 DATA 74,87,151,187,32,12,221,764       [936A]
450 DATA 203,5,198,59,62,4,144,675         [86B0]
460 DATA 95,221,78,7,237,83,107,828        [2D24]
470 DATA 141,81,151,184,32,7,58,654        [9908]
480 DATA 107,141,71,195,39,138,62,753      [6CDC]
```

Listing 2. Maschinencode bringt Geschwindigkeit

```
490 DATA 1,184,32,11,122,230,119,699          [E15A]
500 DATA 87,221,126,9,230,136,24,833          [E058]
510 DATA 25,62,2,184,32,11,122,438            [457A]
520 DATA 230,51,87,221,126,9,230,954          [1A5A]
530 DATA 204,24,9,122,230,17,87,693           [D7FC]
540 DATA 221,126,9,230,238,178,119,1121       [EC80]
550 DATA 221,203,5,70,194,100,138,931         [D9AC]
560 DATA 35,58,107,141,214,4,128,687          [EB6E]
570 DATA 71,81,195,51,138,237,91,864          [7888]
580 DATA 96,146,58,120,146,111,38,715         [B8D6]
590 DATA 0,205,29,188,17,80,141,660           [AD02]
600 DATA 1,96,141,237,160,3,237,875           [0500]
610 DATA 160,3,43,43,205,38,188,680           [CFFC]
620 DATA 235,183,237,66,9,235,56,1021         [43CC]
630 DATA 237,201,42,96,146,203,37,962         [F9CC]
640 DATA 203,20,34,44,179,58,120,658          [F064]
650 DATA 146,111,38,0,203,37,203,738          [9F50]
660 DATA 20,34,46,179,205,25,189,698          [D28A]
670 DATA 62,203,205,252,187,201,237,1347
                                              [63EC]
680 DATA 91,96,146,58,120,146,111,768         [F5E6]
690 DATA 38,0,205,29,188,17,80,557            [5CB8]
700 DATA 141,235,1,96,141,237,160,1011        [7502]
710 DATA 3,237,160,3,235,43,43,724            [D788]
720 DATA 205,38,188,235,183,237,66,1152       [03A2]
730 DATA 9,56,237,201,6,4,33,546              [2DD4]
740 DATA 102,146,54,0,35,16,251,604           [D1E2]
750 DATA 6,1,33,115,141,34,111,441            [316E]
760 DATA 141,14,0,17,108,146,237,663          [9E5A]
770 DATA 83,107,141,26,19,18,126,520          [EB60]
780 DATA 27,18,126,33,106,146,95,551          [B172]
790 DATA 22,0,151,237,82,52,42,586            [999E]
800 DATA 111,141,17,30,0,151,25,475           [DDD0]
810 DATA 34,111,141,12,62,42,185,587          [7454]
820 DATA 200,17,6,0,151,25,126,525            [2876]
830 DATA 184,40,17,4,237,91,107,680           [8304]
840 DATA 141,19,19,19,237,83,107,625          [3D74]
850 DATA 141,42,111,141,24,197,42,698         [6EC8]
860 DATA 111,141,237,91,107,141,26,854        [0D24]
870 DATA 190,40,192,151,18,24,188,803         [5FD2]
880 DATA 33,95,146,54,0,237,115,680           [A20E]
890 DATA 113,141,221,110,2,221,102,910        [EDEA]
900 DATA 3,253,33,99,141,253,54,836           [FF0C]
910 DATA 0,0,126,229,33,32,141,561            [3876]
920 DATA 253,94,0,22,0,25,119,513             [EA1E]
930 DATA 253,52,0,225,17,9,0,556              [82CA]
940 DATA 183,237,82,126,183,40,74,925         [59E2]
950 DATA 221,190,0,32,10,33,95,581            [0A86]
960 DATA 146,54,255,237,123,113,141,1069
                                              [48F4]
970 DATA 201,35,229,33,32,141,253,924         [4FC0]
980 DATA 70,0,190,32,8,225,120,645            [9890]
990 DATA 254,1,40,42,24,218,35,614            [9B94]
1000 DATA 16,242,33,131,141,17,30,610         [A882]
1010 DATA 0,6,42,190,32,18,17,305             [1B08]
1020 DATA 16,0,183,237,82,221,126,865         [68B6]
1030 DATA 4,190,32,3,25,24,167,445            [B280]
1040 DATA 225,24,186,25,16,232,237,945        [3928]
1050 DATA 123,113,141,201,253,126,0,957       [CC60]
1060 DATA 254,0,200,225,61,253,119,1112       [5E58]
1070 DATA 0,24,165,0,0,0,0,189                [7DD4]
1080 DATA 1000                                [B5C8]
2000 REM Maschinencode in den Speicher b
     ringen                                   [067A]
2010 REM                                      [348E]
2020 MEMORY &8998:adr=&8999:laenge=0:zei
     le=10                                    [1E40]
2030 RESTORE                                  [5312]
2040 FOR i=1 TO 7                             [04BC]
2050  READ byte                               [DCEE]
2060  IF byte=1000 THEN 2130                  [693C]
2070  POKE adr,byte:adr=adr+1:laenge=lae
     nge+1:summe=summe+byte                   [D0FE]
2080 NEXT i                                   [0964]
2090 READ qsum                                [37DA]
2100 IF summe<>qsum THEN PRINT"Fehler in
      Zeile";zeile:END                        [2108]
2110 zeile=zeile+10:summe=0                   [AE5A]
2120 GOTO 2040                                [9608]
2130 SAVE "risikomc.obj",b,&8999,laenge       [663C]
2140 END                                      [0C7C]
```

Listing 2. Maschinencode bringt Geschwindigkeit (Schluß)

```
40 REM Laenderdaten (37211-37470)             [89E0]
50 DATA 0,0,20,36,0,194,1,240,255,0,0,0,
   0,0,0,0,239,Alaska,1                       [C03E]
60 DATA 0,0,50,84,0,182,1,239,235,251,75
   ,0,0,0,0,0,255,Kanada,1                    [9C7C]
70 DATA 0,0,48,58,0,152,1,255,190,0,0,0,
   0,0,0,0,251,USA,0                          [7384]
80 DATA 0,0,20,38,0,120,1,251,250,0,0,0,
   0,0,0,0,190,Mexiko,1                       [2A64]
90 DATA 0,0,9,74,0,103,1,190,187,0,0,0,0
   ,0,0,0,250,Mittelamer.,1                   [FED0]
100 DATA 0,0,15,94,0,91,1,250,234,197,25
    4,0,0,0,0,0,187,Kolumbien,1               [ECA8]
110 DATA 0,0,10,112,0,95,1,187,197,0,0,0
    ,0,0,0,0,234,Venezuela,1                  [0B64]
120 DATA 0,0,16,90,0,75,1,187,197,174,17
    5,0,0,0,0,0,254,Peru,1                    [42A8]
130 DATA 0,0,35,146,0,61,1,234,187,254,1
    74,238,26,0,0,0,197,Brasilien,1           [5542]
140 DATA 0,0,15,118,0,62,1,254,197,238,1
    75,0,0,0,0,0,174,Bolivien,1               [AE36]
150 DATA 0,0,9,110,0,39,1,254,174,238,0,
    0,0,0,0,0,175,Chile,1                     [604E]
160 DATA 0,0,21,122,0,39,1,175,174,197,0
    ,0,0,0,0,0,238,Argentinien,1              [6FD4]
170 DATA 0,0,24,214,0,195,1,255,69,0,0,0
    ,0,0,0,0,235,Groenland,1                  [5A4E]
180 DATA 0,0,17,295,185,2,240,85,80,235,
    0,0,0,0,0,69,Skandinavien,1               [E30E]
190 DATA 0,0,7,272,167,2,69,85,0,0,0,0,0
    ,0,0,80,England,1                         [6CC0]
200 DATA 0,0,22,282,157,2,240,84,90,78,8
    1,80,69,0,0,85,Mitteleuropa,1             [5734]
210 DATA 0,0,16,255,0,147,2,75,85,0,0,0,
    0,0,0,0,81,Westeuropa,1                   [BCB0]
220 DATA 0,0,18,330,146,2,240,181,241,85
    ,0,0,0,0,0,84,Tuerkei,0                   [043E]
230 DATA 0,0,25,232,0,132,3,255,81,78,95
    ,0,0,0,0,0,75,Marokko,1                   [0842]
240 DATA 0,0,22,268,128,3,75,85,90,30,26
    ,95,0,0,0,78,Algerien,1                   [67C0]
250 DATA 0,0,24,300,128,3,78,85,27,14,10
    ,30,0,0,0,90,Libyen,1                     [65FE]
260 DATA 0,0,14,224,0,116,3,75,78,26,0,0
    ,0,0,0,0,95,Mauretanien,1                 [B9BA]
270 DATA 0,0,30,230,0,100,3,95,78,30,31,
    197,0,0,0,0,26,Mali,1                     [11F0]
280 DATA 0,0,14,272,110,3,26,78,90,10,31
    ,0,0,0,0,30,Niger,2                       [07A8]
290 DATA 0,0,10,296,108,3,30,90,14,15,31
    ,0,0,0,0,10,Tschad,2                      [F468]
300 DATA 0,0,16,264,98,3,26,30,10,15,0,0
    ,0,0,0,31,Nigeria,1                       [B76C]
310 DATA 0,0,31,296,90,3,31,10,14,91,79,
    94,0,0,0,15,Zaire,1                       [66DC]
320 DATA 0,0,21,288,72,3,15,79,0,0,0,0,0
    ,0,0,94,Angola,1                          [57E4]
330 DATA 0,0,30,298,50,3,94,15,91,0,0,0,
    0,0,0,79,Suedafrika,1                     [0AAE]
340 DATA 0,0,19,320,128,3,90,241,14,0,0,
    0,0,0,0,27,Aegypten,1                     [38CC]
350 DATA 0,0,22,324,108,3,10,90,27,91,15
    ,0,0,0,0,14,Sudan,2                       [58B8]
360 DATA 0,0,26,340,88,3,15,14,241,79,0,
    0,0,0,0,91,Ostafrika,1                    [DCC4]
370 DATA 0,0,100,408,184,4,69,85,84,181,
    164,245,160,239,0,240,UdSSR,0             [9164]
380 DATA 0,0,25,346,128,4,27,84,181,164,
    91,0,0,0,0,241,Saudiarabien,1             [85B0]
390 DATA 0,0,19,368,138,4,241,84,240,164
    ,0,0,0,0,0,181,Iran,2                     [682E]
400 DATA 0,0,32,406,134,4,181,240,245,22
    5,241,0,0,0,0,164,Indien,1                [A4C6]
410 DATA 0,0,21,460,158,4,240,245,0,0,0,
    0,0,0,0,160,Mongolei,0                    [2818]
420 DATA 0,0,45,472,142,4,240,160,180,22
    5,164,0,0,0,0,245,China,1                 [2CEC]
430 DATA 0,0,8,546,146,4,245,0,0,0,0,0,0
    ,0,0,180,Japan,1                          [3864]
440 DATA 0,0,20,484,116,4,164,245,229,0,
    0,0,0,0,0,225,Thailand,1                  [9CDC]
450 DATA 0,0,24,514,88,4,225,244,0,0,0,0
    ,0,0,0,229,Indonesien,1                   [8F80]
460 DATA 0,0,44,556,46,4,229,0,0,0,0,0,0
    ,0,0,244,Australien,1                     [D7CC]
470 DATA 0,160,0,0,0,0,0,0,0,0,0,5,0,0
    ,3,0,0,6,0,0,4,0,0,0,103,0,0,200          [95E8]
480 DATA 0,0,0,0,0                            [532C]
3490 REM Bildschirmdaten (37505-42563) *
     *******                                  [C92E]
3500 DATA 192,0,64,235,138,0,32,240,214,
     0,                                       [1470]
3510 DATA 158,0,3,255,32,0,57,255,105,0,
     4,240,34,0,34,240,213,0,                 [9DBE]
3520 DATA 44,0,3,239,6,0,5,239,60,0,3,25
     5,36,0,3,255,31,0,59,235,101,0           [FF1A]
3530 DATA 8,240,34,0,31,240,60,0,22,240,
     134,0,                                   [9ED8]
3540 DATA 28,0,8,255,7,0,22,255,47,0,9,2
     55,35,0,3,255,3,0,3,255,27,0             [B772]
3550 DATA 3,255,7,0,51,255,97,0,9,240,21
     ,0,3,240,8,0,49,240,10,0                 [C6D4]
3560 DATA 20,240,14,0,71,240,85,0,            [F060]
3570 DATA 28,0,8,239,6,0,23,239,47,0,11,
     255,32,0,4,255,3,0,3,255,39,0            [70D4]
3580 DATA 48,235,98,0,3,240,18,0,4,240,6
     ,0,3,240,7,0,164,240,85,0,               [3CA2]
3590 DATA 24,0,52,255,40,0,6,255,33,0,7,
     255,44,0,47,255,118,0                    [46D0]
```

```
3600 DATA 4,240,6,0,4,240,3,0,170,240,82
     ,0,                                  [E1B0]
3610 DATA 24,0,57,239,31,0,12,255,31,0,1
     5,255,36,0,44,235,121,0              [C770]
3620 DATA 6,240,4,0,176,240,83,0,         [A808]
3630 DATA 29,0,53,255,14,0,12,255,29,0,3
     ,255,14,0,14,255,37,0,46,255         [0E7A]
3640 DATA 98,0,6,240,14,0,8,240,5,0,167,
     240,91,0,                            [8824]
3650 DATA 32,0,44,239,33,255,28,0,3,255,
     16,0,17,255,29,0,46,235,101,0        [47DC]
3660 DATA 9,240,13,0,7,240,6,0,166,240,9
     0,0,                                 [2F36]
3670 DATA 30,0,77,255,8,0,3,255,19,0,3,2
     55,19,0,12,255,35,0,38,255           [C2C6]
3680 DATA 127,0,9,240,4,0,168,240,88,0,   [F76A]
3690 DATA 24,15,48,239,38,255,5,0,9,255,
     11,0,4,255,12,0,3,255,8,0,12,255     [8D14]
3700 DATA 29,0,43,235,126,0,9,240,4,0,16
     8,240,87,15,                         [A05C]
3710 DATA 21,15,89,255,3,0,15,255,6,0,5,
     255,7,0,7,255,11,0,12,255,27,0       [862E]
3720 DATA 44,255,51,0,10,69,53,0,3,240,1
     0,0,9,240,3,0,167,240,87,15,         [BC5A]
3730 DATA 21,0,47,239,73,255,3,0,10,255,
     10,0,10,255,33,0,30,235,60,0         [B048]
3740 DATA 22,69,44,0,3,240,9,0,180,240,8
     5,0,                                 [2542]
3750 DATA 21,0,118,255,4,0,8,255,12,0,15
     ,255,23,0,41,255,54,0,24,69          [930E]
3760 DATA 43,0,5,240,7,0,179,240,86,0,    [5300]
3770 DATA 19,0,45,239,87,255,11,0,21,255
     ,20,15,44,235,48,0,23,69             [031C]
3780 DATA 12,240,7,0,7,240,10,0,4,240,5,
     0,190,240,87,0,                      [8750]
3790 DATA 13,0,131,255,15,0,23,255,24,15
     ,27,255,62,0,23,69,16,240            [E25E]
3800 DATA 4,0,215,240,87,0,               [7EBC]
3810 DATA 14,0,46,239,77,255,16,0,17,255
     ,4,0,8,255,22,0,26,235,14,0          [EB1E]
3820 DATA 3,235,47,0,23,69,18,240,3,0,21
     5,240,87,0,                          [35F4]
3830 DATA 18,0,120,255,10,0,23,255,3,0,3
     ,255,28,0,24,255,15,0,9,255          [CCFE]
3840 DATA 41,0,23,69,17,240,3,0,212,240,
     91,0,                                [6B90]
3850 DATA 14,0,44,239,84,255,8,0,21,255,
     32,0,24,235,16,0,11,235,39,15        [A1E4]
3860 DATA 12,69,4,0,8,69,5,240,15,0,196,
     240,9,0,6,240,92,0,                  [150E]
3870 DATA 11,0,129,255,21,0,3,255,3,0,6,
     255,31,0,20,255,20,0,10,255          [17E4]
3880 DATA 39,15,12,69,3,0,9,69,6,240,13,
     0,198,240,10,0,5,240,91,0,           [9AB4]
3890 DATA 16,0,4,239,22,0,15,239,77,255,
     27,0,3,255,3,0,5,255,30,0,16,235     [E70A]
3900 DATA 25,0,12,235,37,0,12,69,5,0,8,6
     9,206,240,21,0,6,240,90,0,           [7F90]
3910 DATA 14,0,5,255,25,0,78,255,40,0,3,
     255,3,0,4,255,30,0,12,255,30,0       [830C]
3920 DATA 8,255,39,0,12,69,3,0,12,69,204
     ,240,21,0,8,240,89,0,                [A5CA]
3930 DATA 5,0,11,255,37,0,73,255,24,0,7,
     255,5,0,3,255,39,0,10,235,77,0,11,6
     9                                    [7D86]
3940 DATA 5,0,11,69,197,240,28,0,9,240,8
     8,0,                                 [8A60]
3950 DATA 53,0,72,255,23,0,8,255,49,0,8,
     255,79,0,8,69,5,0,14,69,194,240      [36E4]
3960 DATA 30,0,10,240,87,0,               [2BBA]
3970 DATA 52,0,72,255,24,0,8,255,49,0,7,
     235,78,0,9,69,5,0,15,69              [E3CC]
3980 DATA 194,240,31,0,11,240,85,0,       [26D6]
3990 DATA 52,0,71,255,24,0,9,255,53,0,3,
     255,77,0,10,69,4,0,16,69             [A610]
4000 DATA 194,240,31,0,11,240,85,0,       [55B6]
4010 DATA 52,0,72,255,22,0,10,255,131,0,
     11,69,7,0,14,69,196,240              [437E]
4020 DATA 30,0,11,240,84,0,               [079E]
4030 DATA 51,0,73,255,18,0,13,255,133,0,
     10,69,6,0,14,69,196,240              [7192]
4040 DATA 32,0,11,240,83,0,               [2AA4]
4050 DATA 51,0,73,255,17,0,25,255,111,0,
     6,80,3,0,11,69,9,0,12,69             [19DC]
4060 DATA 194,240,35,0,11,240,82,0,       [D2C4]
4070 DATA 51,0,74,255,15,0,27,255,109,0,
     6,80,5,15,10,69,19,0,198,240         [928C]
4080 DATA 35,0,7,240,84,0,                [D35E]
4090 DATA 48,0,77,255,14,0,28,255,103,0,
     3,80,3,0,5,80,7,15,10,69,15,0        [9CCC]
4100 DATA 201,240,16,0,6,240,17,0,3,240,
     84,0,                                [8070]
4110 DATA 50,0,75,255,17,0,27,255,99,0,4
     ,80,3,0,8,80,4,0,11,69,13,0          [9300]
4120 DATA 201,240,20,0,5,240,103,0,       [CF96]
4130 DATA 47,0,79,255,10,0,33,255,98,0,5
     ,80,3,0,8,80,5,0,3,69,3,0            [0D46]
4140 DATA 3,69,11,0,183,240,14,245,22,24
     0,6,0,7,240,100,0,                   [676E]
4150 DATA 46,0,82,255,6,0,39,255,102,0,9
     ,80,8,0,4,69,11,0,223,240            [BC48]
4160 DATA 4,0,4,240,102,0,                [613C]
4170 DATA 46,0,81,255,7,0,36,255,106,0,8
     ,80,10,0,3,15,5,0,189,240,17,245     [24FC]
4180 DATA 23,240,3,0,5,240,101,0,         [9DE4]
4190 DATA 48,0,79,255,6,0,34,255,106,0,1
     0,80,11,0,3,15,4,0,230,240,5,0       [1408]
4200 DATA 7,240,97,0,                     [867A]
4210 DATA 47,0,129,255,96,0,9,80,6,0,12,
     85,192,240,18,245,23,240             [350E]
4220 DATA 5,0,3,240,100,0,                [6D32]
4230 DATA 47,0,61,255,5,0,63,255,96,0,8,
     80,7,0,12,85,233,240                 [1476]
4240 DATA 5,0,3,240,100,0,                [FF36]
4250 DATA 46,0,60,251,8,0,56,255,6,15,98
     ,0,6,80,7,15,13,85,168,240           [6ADC]
4260 DATA 8,160,15,240,22,245,20,240,107
     ,0,                                  [A9D0]
4270 DATA 46,0,63,255,5,0,54,255,6,0,4,1
     5,93,0,8,80,3,15,19,85,232,240       [A74A]
4280 DATA 107,0,                          [D990]
4290 DATA 42,0,68,251,5,0,49,255,12,0,3,
     15,105,0,17,85,131,240,59,160        [0DDC]
4300 DATA 36,245,5,240,108,0,             [2482]
4310 DATA 45,0,67,255,3,0,49,255,14,0,3,
     15,97,0,24,85,230,240,108,0,         [756A]
4320 DATA 43,0,61,251,11,0,47,255,16,0,3
     ,15,93,0,29,85,127,240,67,160        [6ADC]
4330 DATA 31,245,5,240,14,0,3,240,90,0,   [863A]
4340 DATA 41,0,65,255,9,0,43,255,21,0,3,
     15,93,0,28,85,230,240,14,0           [09A0]
4350 DATA 7,240,86,0,                     [64B2]
4360 DATA 39,0,66,251,5,0,46,255,11,0,8,
     255,5,0,3,15,89,0,32,85,126,240      [96A8]
4370 DATA 8,245,59,160,31,245,5,240,14,0
     ,7,180,86,0,                         [1E6A]
4380 DATA 37,0,70,255,3,0,38,255,16,0,8,
     255,9,0,3,15,91,0,29,85              [CEA6]
4390 DATA 229,240,16,0,4,180,87,0,        [8F80]
4400 DATA 35,0,66,251,81,0,3,15,91,0,29,
     85,58,240,6,0,57,240,18,245          [562A]
4410 DATA 48,160,37,245,4,240,16,0,4,180
     ,87,0,                               [5910]
4420 DATA 30,0,116,255,37,0,3,15,90,0,42
     ,85,11,240,9,0,25,240,7,0,163,240    [E836]
4430 DATA 16,0,3,180,88,0,                [0260]
4440 DATA 29,0,119,251,36,0,3,15,88,0,44
     ,85,9,84,10,0,23,240,9,0,56,240      [0DA4]
4450 DATA 21,245,42,160,40,245,111,0,     [7088]
4460 DATA 28,0,121,255,36,0,3,15,68,0,4,
     81,15,0,43,85,5,84,16,0,20,240       [192A]
4470 DATA 13,0,156,240,112,0,             [4776]
4480 DATA 27,0,126,251,33,0,3,15,67,0,9,
     81,10,0,40,85,7,84,18,0,19,240       [4838]
4490 DATA 14,0,54,240,33,245,28,160,40,2
     45,20,0,3,180,89,0,                  [51FE]
4500 DATA 25,0,111,255,8,0,10,255,33,0,3
     ,15,67,0,9,81,7,0,42,85,6,84,24,0    [1B44]
4510 DATA 14,240,14,0,156,240,19,0,4,180
     ,88,0,                               [8BFA]
4520 DATA 25,0,108,251,55,0,3,15,66,0,17
     ,81,12,85,3,0,25,85,6,84,27,0,13,24
     0                                    [827A]
4530 DATA 12,0,52,240,50,245,13,160,24,2
     45,7,0,12,245,19,0,5,180,86,0,       [701A]
4540 DATA 23,0,108,255,58,0,3,15,65,0,18
     ,81,4,85,3,0,4,85,3,0,25,85,6,84,16
     ,0                                   [76EA]
4550 DATA 14,84,11,240,11,0,139,240,10,0
     ,10,240,17,0,5,180,87,0,             [AEB6]
4560 DATA 23,0,108,251,59,0,3,15,64,0,19
     ,81,3,85,4,0,4,85,4,0,14,85,10,0,38
     ,84                                  [2D44]
4570 DATA 8,240,12,0,50,240,87,245,12,0,
     11,245,14,0,8,180,85,0,              [8D76]
4580 DATA 24,0,100,255,67,0,3,15,61,0,22
     ,81,7,0,4,85,5,0,10,85,13,0,39,84    [DB46]
4590 DATA 8,240,10,0,137,240,14,0,10,240
     ,12,0,9,180,85,0,                    [C314]
4600 DATA 23,0,95,251,74,0,3,15,60,0,23,
     81,6,0,4,85,6,0,9,85,15,0,39,84,7,2
     40                                   [0DE2]
4610 DATA 9,0,49,240,90,245,14,0,7,245,1
     2,0,9,180,86,0,                      [A87E]
4620 DATA 23,0,95,255,75,0,3,15,60,0,21,
     81,9,0,3,85,8,0,6,85,15,0,39,84,7,2
     40                                   [0CEE]
4630 DATA 9,0,142,240,13,0,5,240,10,0,12
     ,180,85,0,                           [F35C]
4640 DATA 23,0,95,251,76,0,3,15,57,0,19,
     81,11,0,5,85,9,0,6,85,15,0,39,84,6,
     181                                  [6764]
4650 DATA 10,0,8,181,39,240,95,245,13,0,
     4,240,9,15,11,180,87,0,              [A58C]
4660 DATA 25,0,92,255,78,0,3,15,55,0,18,
     81,13,0,3,15,14,0,3,85,16,0,38,84,8
     ,240                                 [C5AC]
4670 DATA 10,0,140,240,14,0,3,240,9,15,1
     1,180,87,0,                          [36CC]
4680 DATA 25,0,90,251,81,0,3,15,67,0,3,1
```

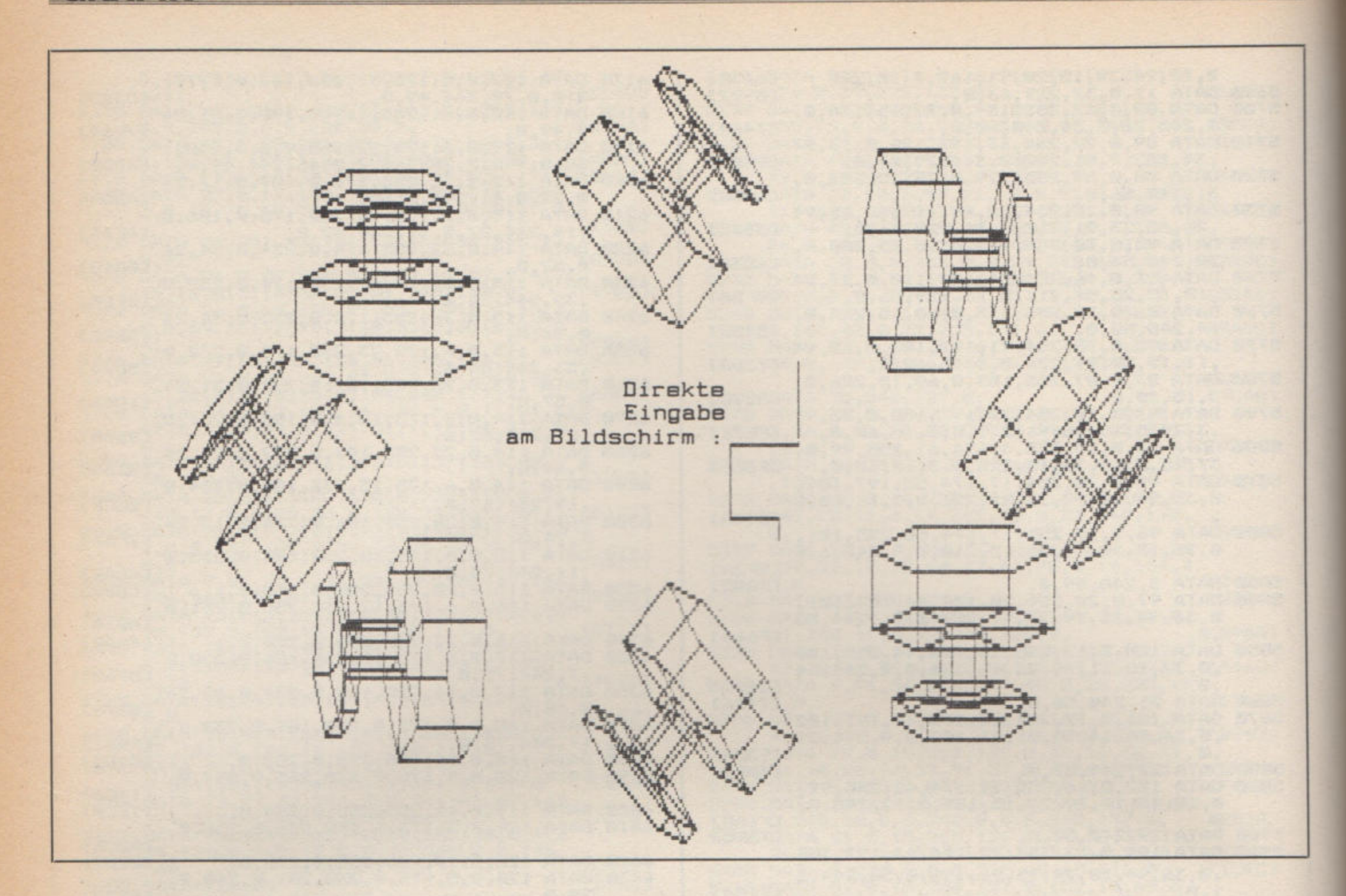
Listing 3. Daten für tolle Grafik

```
     5,15,0,3,15,12,0,4,85,27,0,18,84,10
     ,241                                [8D82]
4690 DATA 31,181,8,240,24,164,96,245,14,
     0,3,240,8,0,9,180,89,0,             [ECAA]
4700 DATA 25,0,90,255,82,0,3,15,66,0,3,1
     5,15,0,3,15,13,0,3,85,28,0,17,84    [5CDC]
4710 DATA 172,240,24,0,5,180,91,0,       [2E68]
4720 DATA 25,0,13,190,75,251,85,0,3,15,5
     3,0,15,75,13,78,4,15,14,0,3,85,37,0
                                         [1320]
4730 DATA 20,241,30,181,33,164,99,245,11
     8,0,                                [A95C]
4740 DATA 26,0,87,255,86,0,3,15,37,0,46,
     15,16,0,3,15,36,0,183,240,117,0,    [9FF2]
4750 DATA 26,0,14,190,71,251,89,0,3,15,3
     8,0,29,75,10,78,21,0,3,15,35,0,23,2
     41                                  [B0E2]
4760 DATA 29,181,38,164,95,245,116,0,    [83CE]
4770 DATA 26,0,84,255,91,0,3,15,35,0,40,
     15,22,0,3,15,35,0,184,240,117,0,    [BAD4]
4780 DATA 25,0,17,190,46,251,5,0,7,251,1
     02,0,3,15,34,0,31,75,11,78,20,0,3,9
     0                                   [6464]
4790 DATA 34,0,26,241,29,181,37,164,98,2
     45,112,0,                           [375C]
4800 DATA 26,0,61,255,6,0,8,255,102,0,81
     ,15,17,0,5,15,33,0,190,240,111,0,   [E622]
4810 DATA 26,0,3,190,3,0,19,190,21,251,2
     1,0,8,251,103,0,31,15,34,75,21,78,1
     0,0                                 [D012]
4820 DATA 9,90,30,0,26,241,29,181,39,164
     ,96,245,111,0,                      [963A]
4830 DATA 26,0,3,255,3,0,39,255,23,0,6,2
     55,133,0,102,15,194,240,111,0,      [0312]
4840 DATA 25,0,3,190,5,0,18,190,17,251,2
     6,0,5,251,133,0,33,75,23,78,32,90,1
     5,27                                [4EB2]
4850 DATA 30,241,30,181,42,164,93,245,11
     0,0,                                [3148]
4860 DATA 25,0,3,255,5,0,33,255,28,0,5,2
     55,132,0,102,15,197,240,110,0,      [1818]
4870 DATA 25,0,4,190,4,0,19,190,14,251,2
     8,0,4,251,133,0,33,75,23,78,33,90   [E66A]
4880 DATA 15,27,30,241,7,0,24,181,48,164
     ,23,245,4,225,59,245,110,0,         [232A]
4890 DATA 26,0,3,255,7,0,28,255,30,0,4,2
     55,135,0,103,15,5,0,25,240,8,0,156,
     240                                 [C332]
4900 DATA 110,0,                         [D282]
4910 DATA 26,0,3,190,5,0,20,190,9,251,30
     ,0,4,251,124,0,42,75,24,78,33,90,16
     ,27                                 [BB3E]
4920 DATA 5,0,26,241,19,0,11,181,59,164,
     8,245,11,225,54,245,111,0,          [E9B2]
4930 DATA 26,0,3,255,5,0,29,255,32,0,3,2
     55,122,0,116,15,6,0,26,240,24,0,137
     ,240                                [E786]
4940 DATA 111,0,                         [A38C]
4950 DATA 34,0,22,190,6,251,157,0,39,75,
     29,78,33,90,17,27,6,0,26,241,26,0   [378E]
4960 DATA 60,164,27,225,47,245,111,0,    [C7B2]
4970 DATA 34,0,28,255,156,0,119,15,6,0,2
     8,240,27,0,131,240,111,0,           [7A3A]
4980 DATA 33,0,28,190,153,0,39,75,34,78,
     34,90,17,27,6,0,27,241,12,0,7,241,1
     3,0                                 [8F60]
4990 DATA 52,164,30,225,43,245,112,0,    [9BA8]
5000 DATA 33,0,28,255,154,0,123,15,6,0,2
     9,240,9,0,13,240,12,0,120,240,113,0
     ,                                   [EF28]
5010 DATA 36,0,25,190,153,0,38,75,36,78,
     33,90,18,27,6,0,30,241,5,0,16,241,1
     6,0                                 [AC42]
5020 DATA 43,164,32,225,37,245,116,0,    [1B9C]
5030 DATA 35,0,27,255,151,0,127,15,5,0,5
     1,240,21,0,96,240,127,0,            [B4C2]
5040 DATA 34,0,28,190,153,0,5,75,22,95,5
     0,78,29,90,20,27,6,0,49,241,21,0,38
     ,164                                [68C8]
5050 DATA 32,225,5,245,7,225,9,245,132,0
     ,                                   [2728]
5060 DATA 34,0,29,255,13,0,10,255,129,0,
     126,15,8,0,47,240,21,0,38,240,10,0  [1092]
5070 DATA 41,240,134,0,                  [3B32]
5080 DATA 33,0,30,190,13,0,4,190,133,0,3
     1,95,6,26,36,78,9,30,26,90,21,14,9,
     0                                   [707A]
5090 DATA 45,241,24,0,33,164,15,0,38,225
     ,134,0,                             [4F62]
5100 DATA 34,0,29,255,11,0,7,255,131,0,1
     31,15,11,0,41,240,24,0,34,240,17,0  [FE6E]
5110 DATA 36,240,134,0,                  [BD30]
5120 DATA 35,0,31,190,6,0,9,190,134,0,28
     ,95,11,26,26,78,16,30,8,10,18,90,22
     ,14                                 [0748]
5130 DATA 13,0,37,241,30,0,27,164,20,0,3
     4,225,135,0,                        [5424]
5140 DATA 36,0,44,255,136,0,130,15,12,0,
     32,240,36,0,23,240,24,0,32,240,135,
     0,                                  [AC92]
5150 DATA 37,0,42,190,17,0,7,250,112,0,2
     8,95,17,26,16,78,19,30,17,10,10,90  [23C6]
5160 DATA 24,14,14,0,26,241,40,15,21,164
     ,26,0,34,225,133,0,                 [EED8]
5170 DATA 40,0,40,255,14,0,10,250,111,0,
     135,15,9,0,24,240,43,15,21,240      [3CF2]
5180 DATA 27,0,32,240,134,0,             [EC18]
5190 DATA 40,0,40,190,7,250,11,0,9,250,1
     07,0,29,95,22,26,6,78,24,30         [4814]
5200 DATA 23,10,5,90,21,14,8,91,7,0,25,2
     41,42,0,20,164,30,0,33,225,131,0,   [440C]
5210 DATA 45,0,35,255,6,250,17,0,7,250,1
     04,0,141,15,6,0,20,240,46,0,19,240  [FE8E]
5220 DATA 30,0,4,240,5,0,24,240,131,0,   [20BC]
5230 DATA 47,0,32,190,6,250,16,0,8,250,1
     05,0,8,26,19,95,21,26,32,30,22,10   [5E46]
5240 DATA 27,14,14,91,3,15,19,241,48,0,1
     7,164,42,0,23,225,131,0,            [9AD8]
5250 DATA 56,0,23,255,6,250,17,0,7,250,1
     04,0,147,15,16,240,51,0,17,240,42,0
                                         [EEFE]
5260 DATA 25,240,129,0,                  [7C40]
5270 DATA 66,0,18,250,20,0,4,250,104,0,4
     7,26,8,31,26,30,22,10,29,14,13,91,3
     ,0                                  [2AB4]
5280 DATA 10,241,59,0,14,164,44,0,23,225
     ,130,0,                             [8652]
5290 DATA 68,0,16,250,16,0,11,250,100,0,
     147,15,71,0,14,240,44,0,23,240,130,
     0,                                  [6C96]
5300 DATA 75,0,7,250,130,0,46,26,6,31,28
     ,30,22,10,28,14,17,91,70,0,14,164,4
     5,0                                 [8926]
5310 DATA 23,225,129,0,                  [B23A]
5320 DATA 74,0,10,250,134,0,141,15,72,0,
     11,240,47,0,22,240,129,0,           [B90A]
5330 DATA 75,0,11,250,131,0,41,26,10,31,
     12,30,12,31,23,10,26,14,19,91,12,0  [A468]
5340 DATA 7,91,54,0,8,164,48,0,22,225,12
     9,0,                                [1350]
5350 DATA 77,0,13,250,128,0,160,15,55,0,
     8,240,48,0,3,240,4,0,16,240,128,0,  [B392]
5360 DATA 79,0,15,250,125,0,38,26,36,31,
     21,10,26,14,37,91,57,0,6,164,49,0   [8570]
5370 DATA 3,225,6,0,12,225,130,0,        [F4F6]
5380 DATA 78,0,17,250,126,0,156,15,57,0,
     5,240,50,0,3,240,8,0,8,240,132,0,   [2B38]
5390 DATA 81,0,15,250,126,0,34,26,37,31,
     21,10,25,14,37,91,59,0,3,164,50,0   [FB50]
5400 DATA 4,225,9,0,5,225,18,15,116,0,   [1CEC]
5410 DATA 81,0,15,250,127,0,152,15,60,0,
     4,240,50,0,4,240,9,0,24,15,114,0,   [6B08]
5420 DATA 83,0,14,250,8,0,5,187,4,234,10
     ,0,10,234,89,0,33,26,37,31,23,15,25
     ,14                                 [5232]
5430 DATA 34,91,148,0,7,15,110,0,        [DF06]
5440 DATA 94,0,3,250,7,0,6,255,20,234,94
     ,0,149,15,152,0,5,15,110,0,         [A5F0]
5450 DATA 95,0,14,187,20,234,3,197,91,0,
     33,26,35,31,31,15,20,14,29,91,67,0  [F6DA]
5460 DATA 6,164,82,0,7,229,107,0,        [EA20]
5470 DATA 95,0,14,255,20,234,5,255,89,0,
     146,15,72,0,4,240,79,0,9,240,107,0,
                                         [2432]
5480 DATA 95,0,20,187,14,234,9,197,86,15
     ,32,26,30,31,36,15,19,14,28,91,155,
     0                                   [3EC8]
5490 DATA 8,229,108,0,                   [A4F4]
5500 DATA 95,0,20,255,14,234,13,255,107,
     15,9,0,112,15,154,0,7,140,109,0,    [B3DE]
5510 DATA 94,0,21,187,13,234,18,197,82,0
     ,18,26,15,0,20,31,47,15,11,14       [2CDE]
5520 DATA 28,91,156,0,8,229,109,0,       [038A]
5530 DATA 94,0,21,255,13,234,20,255,115,
     0,102,15,156,0,10,240,109,0,        [3246]
5540 DATA 95,0,19,187,13,234,24,197,114,
     0,13,31,55,15,31,91,119,0           [C96E]
5550 DATA 5,229,30.0,12,229,110,0,       [F060]
5560 DATA 93,0,21,255,13,234,25,255,134,
     0,78,15,114,0,11,240,28,0,14,240,10
     9,0,                                [718E]
5570 DATA 93,0,20,187,13,234,27,197,131,
     0,49,15,29,91,113,0,20,229,19,0     [79BE]
5580 DATA 17,229,109,0,                  [6456]
5590 DATA 92,0,21,255,13,234,27,255,127,
     0,80,15,122,0,14,240,17,0,22,240,10
     5,0,                                [2A82]
5600 DATA 91,0,21,187,43,197,123,0,54,15
     ,26,91,124,0,17,229,13,0,21,229,107
     ,0,                                 [D542]
5610 DATA 90,0,65,255,123,0,78,15,124,0,
     22,240,10,0,18,240,110,0,           [9E18]
5620 DATA 89,0,4,254,18,187,45,197,122,0
     ,54,15,23,91,130,0,10,229,16,0      [5454]
5630 DATA 19,229,110,0,                  [E242]
5640 DATA 88,0,68,255,122,0,75,15,155,0,
     22,240,110,0,                       [CEB2]
5650 DATA 88,0,8,254,15,187,43,197,124,0
     ,54,15,20,91,162,0,15,229,7,0       [0608]
5660 DATA 18,229,86,0,                   [4EFE]
5670 DATA 90,0,62,255,125,0,75,15,163,0,
     14,240,13,0,38,240,60,0,            [BAD6]
5680 DATA 88,0,11,254,12,187,41,197,127,
```

```
     0,13,94,39,15,20,91,167,0,10,229        [6238]
5690 DATA 17,0,32,229,63,0,                   [09D2]
5700 DATA 88,0,62,255,129,0,72,15,170,0,
     4,240,21,0,34,240,60,0,                 [746A]
5710 DATA 89,0,20,254,43,197,128,0,18,94
     ,34,15,17,91,200,0,37,229,54,0,         [9EBE]
5720 DATA 88,0,64,255,129,0,70,15,202,0,
     31,240,56,0,                            [DA50]
5730 DATA 90,0,20,254,52,197,120,0,25,94
     ,20,15,23,91,210,0,34,229,46,0,         [538C]
5740 DATA 90,0,70,255,122,0,68,15,208,0,
     28,240,54,0,                            [6854]
5750 DATA 91,0,16,254,56,197,118,0,27,94
     ,18,15,25,91,217,0,16,229,56,0,         [9FD8]
5760 DATA 91,0,77,255,115,0,68,15,225,0,
     14,240,50,0,                            [5458]
5770 DATA 92,0,15,254,71,197,105,0,32,94
     ,11,15,25,91,226,0,3,15,60,0,           [73D8]
5780 DATA 93,0,87,255,103,0,68,15,226,0,
     3,15,60,0,                              [829C]
5790 DATA 93,0,19,254,72,197,100,0,35,94
     ,3,79,5,15,25,91,225,0,3,15,60,0,       [FE7E]
5800 DATA 94,0,25,255,3,174,61,255,99,0,
     37,15,3,79,30,15,225,0,3,15,60,0,       [F66E]
5810 DATA 95,0,20,254,17,174,50,197,101,
     0,35,94,10,79,23,91,226,0,3,15,60,0
     ,                                       [598A]
5820 DATA 96,0,19,255,17,174,50,255,101,
     0,35,15,10,79,25,15,210,0,9,240,6,0
                                             [9F2A]
5830 DATA 3,240,59,0,                         [5882]
5840 DATA 97,0,20,254,20,174,44,197,101,
     0,36,94,11,79,23,91,209,0,21,244,58
     ,0,                                     [F66A]
5850 DATA 100,0,17,255,20,174,44,255,100
     ,0,37,15,11,79,23,15,186,0,9,240,14
     ,0                                      [21E0]
5860 DATA 21,240,58,0,                        [79E6]
5870 DATA 101,0,17,254,20,174,41,197,102
     ,0,36,94,14,79,21,91,189,0,9,244,11
     ,0                                      [E30A]
5880 DATA 22,244,57,0,                        [50F2]
5890 DATA 102,0,16,255,20,174,42,255,99,
     0,38,15,14,79,22,15,186,0,12,240,8,
     0                                       [F1A8]
5900 DATA 29,240,52,0,                        [73E0]
5910 DATA 104,0,13,254,22,174,41,197,100
     ,0,36,94,18,79,15,91,190,0,50,244,5
     1,0,                                    [E1B4]
5920 DATA 106,0,11,255,22,174,41,255,100
     ,0,36,15,18,79,12,15,192,0,53,240,4
     9,0,                                    [A4A2]
5930 DATA 106,0,8,254,25,174,40,197,101,
     0,4,94,52,79,7,91,192,0,59,244,46,0
     ,                                       [03BC]
5940 DATA 110,0,4,255,25,174,40,255,102,
     0,55,79,5,15,193,0,58,240,48,0,         [77AA]
5950 DATA 110,0,9,175,21,174,39,197,103,
     0,59,79,192,0,59,244,48,0,              [B3F6]
5960 DATA 110,0,9,255,21,174,37,255,106,
     0,58,79,187,0,64,240,48,0,              [43E4]
5970 DATA 111,0,7,175,16,174,7,238,36,19
     7,106,0,59,79,181,0,68,244,49,0,        [6556]
5980 DATA 111,0,7,255,16,174,42,255,107,
     0,57,79,180,0,71,240,49,0,              [DED6]
5990 DATA 111,0,7,175,14,174,11,238,34,1
     97,104,0,60,79,176,0,76,244,47,0,       [CA96]
6000 DATA 111,0,8,255,13,174,43,255,107,
     0,59,79,174,0,78,240,47,0,              [63C8]
6010 DATA 112,0,5,175,5,238,9,174,14,238
     ,30,197,107,0,59,79,173,0,76,244,50
     ,0,                                     [4972]
6020 DATA 113,0,11,255,7,174,38,255,114,
     0,57,79,174,0,79,240,47,0,              [7DCC]
6030 DATA 111,0,6,175,30,238,21,197,115,
     0,55,79,175,0,80,244,47,0,              [3CC0]
6040 DATA 111,0,56,255,116,0,52,79,178,0
     ,81,240,46,0,                           [69BC]
6050 DATA 112,0,5,175,35,238,13,197,118,
     0,53,79,176,0,84,244,44,0,              [B1D6]
6060 DATA 112,0,53,255,118,0,53,79,176,0
     ,84,240,44,0,                           [B8C0]
6070 DATA 112,0,5,175,34,238,12,197,119,
     0,52,79,178,0,83,244,45,0,              [74DA]
6080 DATA 113,0,48,255,122,0,52,79,177,0
     ,83,240,45,0,                           [49C4]
6090 DATA 112,0,5,175,32,238,12,197,124,
     0,49,79,176,0,84,244,46,0,              [ADDE]
6100 DATA 112,0,48,255,127,0,46,79,180,0
     ,85,240,42,0,                           [E2B6]
6110 DATA 113,0,5,175,30,238,10,179,131,
     0,44,79,179,0,84,244,44,0,              [1ABE]
6120 DATA 113,0,46,255,133,0,40,79,180,0
     ,85,240,43,0,                           [A7A8]
6130 DATA 113,0,5,175,32,238,7,197,136,0
     ,38,79,182,0,82,244,45,0,               [4174]
6140 DATA 113,0,45,255,137,0,33,79,185,0
     ,81,240,46,0,                           [97BE]
6150 DATA 113,0,5,175,34,238,3,197,140,0
     ,30,79,188,0,80,244,47,0,               [5B66]
6160 DATA 112,0,43,255,141,0,16,79,200,0
     ,81,240,47,0,                           [A69E]
6170 DATA 112,0,6,175,37,238,142,0,5,79,
     210,0,79,244,49,0,                      [D3B8]
6180 DATA 112,0,41,255,167,0,190,0,81,24
     0,49,0,                                 [7E64]
6190 DATA 112,0,6,175,22,238,9,0,3,238,1
     68,0,190,0,28,244,7,0,46,244,49,0,      [4DD0]
6200 DATA 112,0,32,255,176,0,189,0,17,24
     0,22,0,41,240,51,0,                     [A5D4]
6210 DATA 112,0,7,175,26,238,175,0,186,0
     ,14,244,27,0,41,244,52,0,               [4F44]
6220 DATA 114,0,32,255,174,0,234,0,34,24
     0,52,0,                                 [8848]
6230 DATA 113,0,6,175,27,238,174,0,232,0
     ,35,244,53,0,                           [DFB0]
6240 DATA 113,0,34,255,173,0,232,0,34,24
     0,54,0,                                 [5B4C]
6250 DATA 113,0,6,175,23,238,178,0,232,0
     ,33,244,55,0,                           [5DB4]
6260 DATA 113,0,26,255,181,0,232,0,31,24
     0,57,0,                                 [1C50]
6270 DATA 114,0,5,175,18,238,183,0,231,0
     ,27,244,62,0,                           [96B8]
6280 DATA 114,0,23,255,183,0,232,0,24,24
     0,64,0,                                 [8154]
6290 DATA 114,0,6,175,15,238,185,0,234,0
     ,19,244,67,0,                           [0CCE]
6300 DATA 114,0,20,255,186,0,234,0,12,24
     0,74,0,                                 [7F46]
6310 DATA 115,0,5,175,15,238,185,0,235,0
     ,11,244,74,0,                           [6DAE]
6320 DATA 115,0,20,255,255,0,250,0,          [CBBA]
6330 DATA 116,0,6,175,12,238,255,0,251,0
     ,                                       [A01A]
6340 DATA 116,0,17,255,255,0,252,0,          [4AD0]
6350 DATA 117,0,5,175,12,238,186,0,238,0
     ,7,244,75,0,                            [D768]
6360 DATA 117,0,14,255,189,0,234,0,18,24
     0,68,0,                                 [0876]
6370 DATA 118,0,5,175,8,238,189,0,230,0,
     17,244,73,0,                            [F86C]
6380 DATA 118,0,14,255,255,0,253,0,          [D4D8]
6390 DATA 120,0,4,175,7,238,255,0,254,0,
                                             [14C6]
6400 DATA 119,0,12,255,255,0,254,0,          [1ECA]
6410 DATA 119,0,5,175,6,238,255,0,255,0,
                                             [A2CA]
6420 DATA 120,0,10,255,255,0,255,0,          [02BC]
6430 DATA 120,0,5,175,4,238,251,0,240,0,
     20,0,                                   [417A]
6440 DATA 122,0,7,255,251,0,240,0,20,0,      [BB30]
6450 DATA 122,0,4,175,254,0,240,0,20,0,      [8534]
6460 DATA 200,0,200,0,240,0,                  [EDF8]
6470 DATA 129,0,6,238,255,0,250,0,            [1F7A]
6480 DATA 130,0,8,255,252,0,250,0,            [1768]
6490 DATA 200,0,200,0,240,0,                  [50FE]
6500 DATA 200,0,200,0,240,0,                  [B3EE]
6990 END                                      [0B9E]
9000 REM Daten in den Speicher poken ***
     ********                                [7A4A]
9010 MEMORY 36199:RESTORE                     [B19A]
9020 FOR i=36211 TO 37504                     [2DF6]
9030  READ a$                                 [FC9A]
9040  IF a$>="A" THEN a$=a$+SPACE$(12-LE
     N(a$)):GOTO 9090                        [2B5A]
9050  a=VAL(a$)                               [2ACA]
9060  IF a>255 THEN POKE i,a-INT(a/256)*
     256:i=i+1:POKE i,INT(a/256):GOTO 91
     30                                      [9F2C]
9070  POKE i,a                                [3FAA]
9080  GOTO 9130                               [1C6E]
9090  FOR j=1 TO 12                           [2E6E]
9100   POKE i+j-1,ASC(MID$(a$,j,1))           [A7E0]
9110  NEXT j                                  [A4A8]
9120  i=i+11                                  [2A50]
9130 NEXT i                                   [E46A]
9140 adr=37505                                [3DCC]
9145 sum=0                                    [986A]
9150 READ a                                   [4118]
9160 POKE adr,a:adr=adr+1:sum=sum+a           [EFE2]
9170 IF a<>0 THEN READ b:POKE adr,b:adr=
     adr+1:GOTO 9150                         [F852]
9180 IF sum<>640 THEN 10000                   [54D4]
9190 IF adr<42563 THEN 9145                   [A5B6]
9200 SAVE "risiko.bin",b,36200,6364:STOP
                                             [D554]
10000 datadr=PEEK(&AE30)+256*PEEK(&AE31)
                                             [7F76]
10010 IF PEEK(datadr)=0 THEN 10060            [D8F4]
10020 IF PEEK(datadr)<>ASC(",") THEN PRI
      NT"Fehler !!!":STOP                    [C4A6]
10030 IF PEEK(datadr)<>&8C THEN datadr=d
      atadr-1:GOTO 10030                     [CF6A]
10040 zeile=PEEK(datadr-2)+256*PEEK(data
      dr-1)                                  [A4CC]
10050 GOTO 10070                              [66CE]
10060 zeile=PEEK(datadr+3)+256*PEEK(data
      dr+4)-10                               [23EC]
10070 PRINT "Falsche Quersumme in Zeile
      ";zeile                                [3C4E]
10080 END                                     [E6E0]
```

Listing 3. Daten für tolle Grafik (Schluß)





Der 3D-Dreh

Betrachten Sie mit dem Programm »3D-Dreh« beliebige Rotationskörper von allen Seiten. Auf die Perspektive kommt es an!

Ein Rotationskörper im mathematischen Sinn entsteht, wenn eine sogenannte »Mantellinie« eine volle Umdrehung um eine Achse geführt wird. Die Mantellinie erzeugt dabei die Oberfläche des Körpers. Um eine grafisch sinnvolle Darstellung im Drahtmodell zu ermöglichen, muß die Umdrehung in Winkelschritten erfolgen. Außerdem ist die Darstellung der Mantellinie als Streckenzug (Polygonzug) erforderlich.

Um einen Körper in unserem Programm zu editieren, können Sie dessen Mantellinie auf zwei Wegen eingeben:

Bei der Eingabe über DATA-Zeilen werden im Programmtext die Eckpunkte des Streckenzugs programmiert. Für einen Eckpunkt sind drei Daten erforderlich:

1. Nummer des Punktes
2. X-Koordinate
3. Y-Koordinate

Der Bildschirmmittelpunkt ist der Koordinatenursprung (0/0).

Nehmen wir als Beispiel die Darstellung einer vierseitigen Pyramide. Hierzu müssen Sie ein rechtwinkliges Dreieck eingeben. Die Drehachse fällt dabei mit einer der beiden kürzeren Seiten des Dreiecks zusammen, die den rechten Winkel bilden (Katheten). Rotiert nun dieses Dreieck in Winkelschritten von 90 Grad, so entsteht der gewünschte Körper. Tippen Sie zum Beispiel folgende Zeilen ein:

```
270 st=90
840 DATA 1,0,100,2,140,-70,3,0,-70
```

Die Variable »st« in Zeile 270 bestimmt die Größe der Winkelschritte. Wählen Sie st=5, so erhalten Sie einen Kegel. Nach dem Start des Programms müssen Sie bei der Abfrage des Eingabemodus »d« eintippen. Für Drehungen um alle drei Achsen erweist sich ein Wert von 30 Grad in allen Richtungen als sinnvoll.

Komfortabler und schneller gestaltet sich die Eingabe der Mantellinie über den Editor. Sie erreichen ihn im Eingabemodus mit der Taste »b« und verfügen hierbei zur Orientierung über ein Koordinatenkreuz und eine Koordinatenanzeige. Die Cursor-Tasten bewegen einen Pixel-Cursor über die Koordinatenebene. SPACE hält jeweils die aktuelle Cursor-Position fest. Die so gespeicherten Koordinaten werden zur Mantellinie verbunden, der Editor mit ENTER verlassen. Sie bestimmen anschließend noch die drei Winkel für die Perspektive.

Da das Programm selbstdokumentierend ist, können Sie seine Bedienung und Funktionen ohne weitere Anleitung erkunden. (Frank Müller/Matthias Rosin)

Programm-Steckbrief	
Name:	3D-Dreh
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
10 REM                                      [E8CA]
20 REM ******* 3-D-DREH-FIGUREN *******     [1D86]
30 REM ****** VON FRANK MUELLER *******     [3148]
40 REM                                      [01D0]
50 DEG:MODE 1                               [9B0A]
60 KEY 0,CHR$(13)+"mode 2:pen 1"+CHR$(13
   )+"list ":KEY 1,CHR$(13)+"edit "         [DE18]
65 GOTO 890                                 [340A]
70 DIM x(1000),y(1000),z(1000)              [78F4]
80 ORIGIN 320,200                           [F586]
90 INPUT"Eingabe ueber DATAs oder direkt
    am{6 SPACE}Bildschirm (d/b){24 SPACE
   }[Ein Beispiel ist eingegeben]";a$:IF
    a$="b" THEN 630                         [6E7A]
100 REM                                     [5E2A]
110 REM ******* DATAS EINLESEN ********     [4898]
120 REM                                     [5C2E]
130 READ c,x,y:IF c=-9999 THEN 150          [F076]
140 i=i+1:x(i)=x:y(i)=y:GOTO 130            [DE16]
150 z=i                                     [BDAC]
160 CLS                                     [8332]
170 REM                                     [6B38]
180 REM ******* DATEN ZEICHNEN ********     [61A6]
190 REM                                     [A13C]
200 FOR t=1 TO i                            [A9CE]
210 IF t=1 THEN MOVE x(t),y(t)              [E062]
220 DRAW x(t),y(t),1                        [360C]
230 NEXT t                                  [0310]
240 REM                                     [4C34]
250 REM *********** DREHEN ************     [B0EE]
260 REM                                     [5A38]
270 st=60                                   [9686]
280 FOR w=st TO 360 STEP st:an=an+1:FOR
    t=1 TO i:z=z+1:x(z)=COS(w)*x(t):y(z)
    =y(t):z(z)=x(t)*SIN(w):NEXT t,w         [8E5C]
290 s=i:i=z:r=0                             [FC8E]
300 MODE 1                                  [9852]
310 ORIGIN 320,200                          [83DE]
320 REM                                     [6032]
330 REM ******* WINKELEINGABE *********     [E89E]
340 REM                                     [6236]
350 INPUT"1.Winkel :";w1:INPUT "2.{7 SPA
    CE}:";w2                                [243C]
360 INPUT "3.{7 SPACE}:";w3                 [6A1A]
370 c1=COS(w1)::s1=SIN(w1)                  [6842]
380 CLS                                     [893A]
390 REM                                     [B140]
400 REM ********** ZEICHNEN ***********     [AA7C]
410 REM                                     [6532]
420 PLOT -1000,0,1                          [B178]
430 FOR t=1 TO i                            [ADD8]
440 d=SQR(x(t)*x(t)+z(t)*z(t))              [0B2E]
450 IF x(t)<>0 THEN w4=ATN(z(t)/x(t)) EL
    SE w4=SGN(z(t))*90                      [0CC2]
460 IF x(t)<0 THEN w4=w4+180                [41B0]
470 y=y(t):x=COS(w3+w4)*d:z=SIN(w3+w4)*d
                                            [CB38]
480 l=SQR(x*x+y*y)                          [B11A]
490 IF x<>0 THEN w5=ATN(y/x):IF x<0 THEN
     w5=w5+180                              [149E]
500 IF x=0 THEN w5=SGN(y)*90                [D2CC]
510 x1=l*COS(w2+w5):y1=l*SIN(w2+w5)*c1+s
    1*z                                     [3716]
520 IF t=1 OR ts=1 OR (t-1)/s=INT((t-1)/
    s) AND ti<>1 THEN MOVE x1,y1            [FA84]
530 DRAW x1,y1                              [F90A]
540 IF r=1 THEN 610                         [B79C]
550 NEXT t                                  [0A1A]
560 REM                                     [603E]
570 REM * VERBINDUNGSLINIEN ZEICHNEN **     [8FF0]
580 REM                                     [3E42]
590 PLOT -1000,0,3                          [8C8C]
600 r=1:FOR ti=1 TO s:FOR ts=1 TO an+1:t
    =(ts-1)*s+ti:GOTO 440                   [84DA]
610 IF r=1 THEN NEXT ts,ti:r=0              [343A]
620 GOTO 350                                [FE52]
630 REM                                     [7B3A]
640 REM ***** BIIDSCHIRM-EINGABE ******     [3184]
650 CLS                                     [243A]
660 MOVE -320,0:DRAW 320,0,1:MOVE 0,-200
    :DRAW 0,200,1                           [320C]
670 GOTO 750                                [9A64]
680 PLOT x,y,3                              [7032]
690 i$=INKEY$:IF i$="" THEN 690             [B064]
700 PLOT x,y,0                              [9D1E]
710 as=ASC(i$)                              [5CFC]
720 x=x+2*(as=242)-2*(as=243):y=y+2*(as=
    241)-2*(as=240)                         [9FAA]
730 IF as=13 THEN i=z:GOTO 150              [E22C]
740 IF i$=" " THEN 770                      [094A]
750 LOCATE 1,1:PRINT x;y;                   [D9E0]
760 GOTO 680                                [DA68]
770 IF z=0 THEN MOVE x,y ELSE MOVE x(z),
    y(z)                                    [5654]
780 DRAW x,y,1                              [690E]
790 z=z+1:x(z)=x:y(z)=y                     [0966]
800 GOTO 680                                [EB5E]
810 REM                                     [7D3A]
820 REM ******** IHRE DATAS **********      [464E]
830 REM                                     [7F3E]
840 DATA 1,100,100,2,100,20,3,26,20,4,26
    ,-52,5,80,-52,6,80,-72                  [D7D8]
850 DATA -9999,0,0                          [7D80]
860 REM                                     [7E44]
870 REM ******** EINLEITUNG **********      [3CD6]
880 REM                                     [6048]
890 PEN 1:INK 2,18                          [B9F0]
900 PRINT"{7 SPACE}3-D-DREH-FIGURENPROGR
    AMM"                                    [3C00]
910 MOVE 102,382:DRAWR 396,0,3:MOVE 102,
    378:DRAWR 396,0,3                       [B49A]
920 PRINT:PRINT:PRINT"Als Eingabe benoet
    igt das Pg nur eine{3 SPACE}Haelfte
    einer Figur.Das Pg dreht dann{3 SPAC
    E}diese Figurenhaelfte um die Y-Achs
    e."                                     [1410]
930 PRINT"Die Eingabe kann wahlweise ueb
    er DATAs{2 SPACE}oder direkt am Bild
    schirm erfolgen.Dabeiist der Ursprun
    g immer in der Bild-{5 SPACE}schirmm
    itte."                                  [A210]
940 PEN 2                                   [FFE4]
950 PRINT"Bei Eingabe ueber DATAs sind 3
     Eingaben fuer einen Eckpunkt erford
    erlich.":PRINT" 1.Nummer des Punktes
    .(beliebig)"                            [1FCE]
960 PRINT" 2.X-Koordinate":PRINT" 3.Y-Ko
    ordinate":PRINT"Die lezte DATA-Zeile
     muss -9999,0,0{5 SPACE}enthalten.(B
    eispiel siehe Zeile 820 ff)"            [C9AA]
970 PEN 3                                   [34EC]
980 PRINT"Bei einer direkten Bildschirme
    ingabe istfolgendes zu beachten :":P
    RINT" 1.Mit den Cursortasten wird ei
    n kleiner{3 SPACE}Punkt gesteuert.Zu
    m Speichern eines{5 SPACE}Punktes wi
    rd die SPACE-TASTE gedr."               [D974]
990 PRINT" 2.Als Abschluss druecken Sie
    die ENTER-{3 SPACE}TASTE."              [A7B2]
1000 GOSUB 1150                             [5A90]
1010 CLS:PRINT"Das Pg erwartet die Einga
     be 3er Winkel.":PRINT:PEN 2:PRINT"
     1.Drehwinkel um die X-Achse."          [E346]
1020 PEN 3:PRINT" 2.Dehwinkel um die Z-A
     chse."                                 [C198]
1030 PEN 1:PRINT" 3.Drehwinkel um die Y-
     Achse."                                [8A7A]
1040 MOVE 250,250:DRAWR 0,60,1:MOVE 250,
     250:DRAWR 60,0,2:MOVE 250,250:DRAWR
      30,30,3                               [C714]
1050 LOCATE 15,6:PRINT"y";:PEN 2:LOCATE
     21,10:PRINT"x";:PEN 3:LOCATE 19,8:P
     RINT"z";                               [4608]
1060 PRINT:PRINT:PRINT:PEN 1                [1DE0]
1070 PRINT:PRINT"Die Variable st in Zeil
     e 270 gibt den{3 SPACE}Gradundersch
     ied zwischen den einzelen{3 SPACE}D
     rehungen an."                          [B374]
1080 PRINT:PRINT"Sollen die roten Verbin
     dungslinien weg- gelassen werden mu
     ss an Zeile 600 ein{3 SPACE}REM ges
     etzt werden."                          [1528]
1090 PRINT:PRINT"Wenn Sie die Instruktio
     nen zur{2 SPACE}Genuege kennen,entf
     ernen Sie einfach Zeile 65."           [ABB8]
1100 GOSUB 1150                             [C792]
1110 PRINT"Mit diesem Pg kann auch eine
     beliebige{2 SPACE}Figur verarbeitet
      werden.Dann sollte{4 SPACE}aber mi
     t DATAs gearbeitet werden.An"          [957C]
1120 PRINT"Zeile 280 muss dann einfach e
     in 'REM'{3 SPACE}gesetzt werden.Ein
     e Drehung um die{6 SPACE}Y-Achse wi
     rd dann nicht mehr durch-{5 SPACE}g
     efuehrt."                              [E2EC]
1125 PRINT:PRINT:PRINT"Am Anfang experim
     entiert man am besten{2 SPACE}mit d
     em bereits in Zeile 840 eingegebe-
     nem Beispiel.So laesst sich die Wir
     kung der einzelnen Winkel am besten
      erkennen."                            [33D8]
1126 PRINT"Probieren Sie z.B. folgende W
     inkel :"                               [BA44]
1127 PRINT"1. 0,0,0":PRINT"2. 0,90,0":PR
     INT"3. 90,0,0":PRINT"4. 10,0,0":PRI
     NT"5. 10,0,45":PRINT"6. 90,80,45":P
     RINT" etc."                            [A170]
1130 GOSUB 1150                             [5998]
1140 GOTO 70                                [724C]
1150 LOCATE 35,25:PEN 1:PRINT"TASTE";:CA
     LL &BB06                               [ACEA]
1160 CLS:RETURN                             [5CC8]
```

Listing. Mit diesem Programm lassen Sie Ihren Schneider rotieren



Malermeister

Titelbilder und Hintergrundgrafiken zaubert »Background-Painter« komfortabel und schnell. Mühevolle »Planspiele« auf Millimeter-Papier und die anschließende Rechenarbeit zur programmtechnischen Umsetzung gehören damit der Vergangenheit an. Lassen Sie Ihrer Kreativität freien Lauf.

Um ein gutes Spiel zu programmieren, muß neben Spielidee und Sound auch die Grafik stimmen. Aber es ist recht mühsam, erst die Hintergrundgrafiken auf Papier zu entwerfen und dann das entsprechende Programm einzutippen. Ärgerlich wird es besonders dann, wenn man nachträglich etwas ausbessern will. Mit dem »Background-Painter« ersparen Sie sich diesen unnötigen Arbeitsaufwand.

Das Programm funktioniert ähnlich einem Construction-Set, jedoch lassen sich auch die einzelnen Bauelemente neu definieren. Die Zeichen haben eine Größe von 4 x 8 Punkten in 16 Farben (Modus 0). Gemäß der Zeichenmatrix passen in eine Zeile 40 und in eine Spalte 25 Zeichen. Folglich haben 1000 Zeichen auf dem Bildschirm Platz. Jedes Bild belegt somit 1000 Byte im Speicher. Hinzu kommt noch die Tabelle für die Zeichendefinition, die bei maximaler Ausnutzung der 256 verschiedenen Zeichen nochmals 4 KByte Speicher einnimmt.

Sie entwerfen zunächst mit »Background Painter« die Grafik und übernehmen dann die automatisch erzeugte Zeichendefinition samt Bildschirmaufbau in Ihr eigenes Programm. Der Vorteil besteht darin, daß Sie nach Herzenslust auf dem Bildschirm herumprobieren können, bis Ihnen das Bild gefällt.

Maschinencode-Routinen für zeitkritische Abläufe

Das Programm besteht aus einem Basic-Teil (Listing 1) und Maschinencode-Routinen (Listing 2) zur Ausgabe des Cursors, Vergrößerung des Bildes, das Auslösen der Garbage-Collection (Speicher-Reorganisation), etc. Den Startadressen sind gleich am Beginn (Listing 1, Zeilen 175 bis 190) Namen zugeordnet, um die Lesbarkeit zu verbessern. Listing 3 enthält die Daten für das Titelbild. Wenn Sie auf das Titelbild als Demonstration verzichten wollen, dürfen Sie Listing 3 weglassen. Die Funktion des Programmes bleibt gewährt.

Die Speicherorganisation:

Adresse	Bereich	
	Basic	
8732 hex	Maschinencode-Routinen	
8E71 hex	Daten für Unterroutinen	
9283 hex	Farbentabelle	Dieser Bereich enthält die Bilddaten für die vollständige Grafik
9293 hex	Bereich für Bilddaten	
967B hex	Bereich für Zeichensatz	
A67B hex		

Wie bauen Sie nun die fertigen Grafiken in Ihr Programm ein? Dazu benötigen Sie die Maschinenroutine aus Listing 4, die die Zeichen aus den vom Background-Painter erstellten Tabellen herausliest und als Grafik auf dem Bildschirm sichtbar macht. Nachdem diese Routine auf Diskette gespeichert ist, läßt sie sich später an jede beliebige Adresse laden. Dann muß man jedoch das höchstwertige Byte der neuen Adresse im 17ten Byte ändern (normalerweise steht dort »&C0«). Der Aufruf erfolgt mit

»`CALL adresse,spielfeld,zeichensatz`«.

adresse ist die Lade-Adresse der Maschinenroutine.

spielfeld (Bereich für Bilddaten) ist die Adresse, ab der sich die Tabelle mit den Zeichennummern befindet. Sie liegt direkt hinter der Farbtabelle (siehe Speicherübersicht).

zeichensatz ist die Adresse, ab der die Zeichendefinition gespeichert ist (befindet sich hinter der Spielfeldtabelle).

Hier nun Schritt für Schritt die Vorgehensweise anhand eines Beispiels:

Die fertige Grafik haben Sie bereits gespeichert. Zuerst schützen Sie den Speicher ab Adresse 7000 hex mit dem Befehl »MEMORY &6FFF« vor Überschreiben und laden dorthin den Maschinencode mit »LOAD "prscreen.bin", &7000«. Die Grafik laden Sie ebenfalls, beispielsweise durch »LOAD "grafik",&8000«. Den ersten Wert im Zeichensatz-Speicher belegt der »Background Painter«. Er muß bei Darstellung der Grafik Null sein, da sonst lauter Punkte in der Grafik erscheinen. Also führen Sie immer ein »POKE &8000+1016,0« aus (8000 hex ist die Basisadresse, dazu kommen 16 Farb-Byte und 1000 Byte für die Spielfeldtabelle). Nun setzen Sie noch die Farben entsprechend den Werten der ersten 16 Byte:

»`FOR i=0 TO 15:INK i,PEEK (&8000+i):NEXT i`«.

Um die Grafik dann aufzubauen, genügt der Aufruf

»`MODE 0:CALL &7000,&8000+16,&8000+1016`«.

Während des Programmlaufs erscheint am linken Bildschirmrand das Menü. Die einzelnen Menüpunkte wählen Sie mit dem Cursor aus, indem Sie ihn auf die Symbole richten und mit Tastendruck die entsprechende Funktion aktivieren. Steuern läßt sich der Cursor wahlweise mit einem Joystick oder den Cursor-Steuertasten und der Copy-Taste als Feuerknopf. Bewegen Sie den Cursor zum unteren Bildschirmrand, scrollt das Menü nach oben und zeigt die restlichen Funktionen an. Angefangen von oben bedeuten die einzelnen Menüpunkte:

Paint: enthält die wichtigsten Funktionen für die Gestaltung der Grafik. Mit dem Cursor bestimmen Sie die Position und setzen das aktuelle Zeichen durch Druck auf den Feuerknopf (Copy-Taste). Dieses Zeichen muß bereits definiert sein. Es läßt sich so lange beliebig oft auf den Bildschirm bringen, bis Sie ein anderes Zeichen als aktuelles Zeichen bestimmen. Das geschieht, indem Sie den Cursor auf ein Zeichen richten und es mit »CTRL-G« anwählen. Ist das gewünschte Zeichen auf dem Bildschirm nicht vorhanden, rufen Sie mit »CTRL-T« die gesamte Zeichensatztabelle auf. Neben diesen beiden stellt »Paint« noch weitere Unterfunktionen bereit:

- **CTRL-T** Aufruf der Zeichensatz-Übersicht.
- **CTRL-G** Zeichen unter dem Cursor wird aktuelles Zeichen.
- **CTRL-N** Ausgabe des ASCII-Codes für das aktuelle Zeichen.
- **CTRL-Z** Umschalten zwischen vergrößerter und normaler Darstellung.
- **CTRL-D** definiert neues Zeichen (Zeichen unter dem Cursor wird in die Definition übernommen, es können also vorhandene Zeichen als Vorlage dienen).

CTRL-R ändert definierte Zeichen

In den letzten beiden Routinen (CTRL-D und CTRL-R) sind folgende Funktionen verfügbar:

CTRL-O überlagert zwei Zeichen (zweites Zeichen hat höhere Priorität).
CTRL-C Farbe für Punkt wählen.
CTRL-B Abbruch der Definition.
CLR löscht Definition.
ENTER übernimmt Definition in die Zeichensatztabelle.
CTRL-Z vergrößert Zeichen bei der Definition.

Nur bei der vergrößerten Zeichendefinition:

CTRL (und Pfeil-Taste hoch/runter) spiegelt an der x-Achse.
CTRL (und Pfeil-Taste rechts/links) spiegelt an der y-Achse.
SHIFT (und Pfeil-Taste) schiebt in jeweilige Richtung.
CTRL-S sucht Zeichen in Zeichensatztabelle.

Die Definition bei vergrößerter Darstellung unterscheidet sich etwas von der normalen Darstellung. Hier wird ein Punkt immer durch Drücken des Feuerknopfes gesetzt und mit »Space« (Leer-Taste) gelöscht.

Color: Im Color-Menü wählen Sie aus der Farbpalette die 16 Farben für Ihre Grafik. Die Wahl der einzelnen Farbtöpfe erfolgt wieder mit Hilfe des Cursors. Die Farb-Nummern sind durch Bewegung des Joysticks (der Cursor-Tasten) nach rechts oder links zu erhöhen beziehungsweise zu erniedrigen. Bei Druck von »CTRL-A« erscheint wieder das Bild, und die Farbe ist direkt im Bild veränderbar. ENTER übernimmt den Farbwert.

Line: zieht zwischen zwei frei wählbaren Punkten eine Linie. Diese wird in Elemente (Zeichen) von 4x8 Punkten zerlegt und im Zeichensatz abgelegt. Das Programm beachtet dabei selbständig, daß ein Zeichen nicht mehrfach im Speicher steht. Die Position der Endpunkte läßt sich mit dem Cursor zunächst grob festlegen. Durch Drücken des Feuerknopfes (der Copytaste) erscheint ein Punkt, dessen Positionierung dann die endgültige Position bestimmt.

CTRL-C legt die Farbe des Punktes fest.
CTRL-B löscht die Positionen.

Circle: funktioniert ähnlich »Line«. Jedoch bestimmt die erste Position den Mittelpunkt, die zweite Position den Radius des Kreises.

Zoom: dient der Vergrößerung des aktuellen Bildschirmsektors. Dieser Sektor umfaßt 5 x 3 Zeichen. In der Mitte befindet sich der Cursor. Das Bild scrollt bei Positionsänderung über den Bildschirm. Bis auf »CTRL-T« und »CTRL-N« haben Sie auch bei vergrößerter Darstellung alle Funktionen aus Menüpunkt »Paint« zur Verfügung.

Copy: kopiert Inhalte von Bildschirmbereichen. Sie markieren den zu kopierenden Bereich mit dem Cursor an der linken oberen und rechten unteren Ecke. Dabei bleibt das Zeichen des Cursors an der jeweiligen Stelle stehen. Nun steht der Bildschirmsektor im Puffer und läßt sich beliebig oft kopieren. Dazu geben Sie durch Drücken des Feuerknopfes die neue Position der linken oberen Ecke an. Mit »Space« wählen Sie einen anderen Bereich.

Move: ist genauso zu bedienen wie »Copy«, jedoch löschen Sie den ursprünglichen Bildschirmbereich.

Fill: füllt einen markierten Bildschirmbereich (siehe »Copy«) mit einem gewählten Zeichen. Mit der Leertaste bestimmen Sie einen neuen Bereich.

Garbage Collection: ist eine Routine, die alte, nicht mehr benötigte Zeichen aus dem Zeichenspeicher entfernt. Dabei werden Zeichen gelöscht, die zwar im Zeichensatz noch definiert sind, aber in der Grafik keine Verwendung mehr finden. So läßt sich nach Fertigstellung einer Grafik der Zeichensatz von überflüssigen Zeichen bereinigen und Speicherplatz sparen.

Clear Screen: löscht den Bildschirm ganz oder teilweise. Beim Löschen von Bildteilen ist wieder nach dem Prinzip »von links-oben nach rechts-unten« der Bereich zu markieren.

Clear Character: löscht ein Zeichen, das Sie nur mit dem Cursor auszuwählen brauchen. Dabei rücken die Zeichen im Zeichensatz um eine Position nach vorn. In das Hauptmenü gelangen Sie, indem Sie das erste Zeichen der Zeichensatztabelle (Leerzeichen) anwählen.

Help: ruft eine Hilfsfunktion auf.

Value: dient zur Ausgabe der zweimal acht Byte (linke Seite und rechte Seite), aus denen sich jedes Zeichen zusammensetzt, auf Bildschirm oder Drucker.

Disc: stellt folgende Funktionen zur Wahl:

load file lädt alles, was Sie zuvor mit »Background Painter« gespeichert haben (Grafik und/oder Zeichensatz). Dabei erkennt das Programm selbständig, ob eine Grafik vollständig oder nur teilweise gespeichert ist. Somit lassen sich beliebig viele Teile zusammen in den Bildschirm laden, wobei Sie die ursprüngliche oder eine neue Position wählen.

save all speichert sowohl den Zeichensatz als auch die Grafik.

save picture speichert nur das Bild, weshalb Sie darauf achten sollten, daß bei wiederholtem Laden sich auch der ursprüngliche Zeichensatz im Speicher befindet (sonst gibt es nur Zeichen-Wirrwarr). Auch einzelne Bildschirmbereiche lassen sich speichern, um diese dann wieder in anderen Bildern zu verwenden.

save table speichert den Zeichensatz oder auch nur einzelne Zeichen. Die Markierungen werden hier, wie auch bei »save picture«, mit dem Cursor festgelegt.

load table append hängt einen allein gespeicherten Zeichensatz an einen schon bestehenden an. Dabei müssen Sie darauf achten, daß Sie die maximale Länge von 255 Zeichen nicht überschreiten.

show picture und **show table** erlauben nach dem Ladevorgang die Betrachtung des Bildes oder der Grafik.

catalog gibt ein Inhaltsverzeichnis des Kassetten-/Disketten-Laufwerks aus.

menu löst die Rückkehr ins Hauptmenü aus (wie die Enter-Taste).

(Peter Ullrich/ja)

Programm-Steckbrief	
Name:	Background Painter
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette



```
10 '        background-painter               [8D50]
20 '   (c) 1985 by peter ullrich             [BEC4]
25 '        felixallee 8                     [B0C2]
26 ' 8482 neustadt                          [F15A]
27 '        09602/1451                       [4DD6]
30 '                                         [5F54]
40 ' ON BREAK GOSUB 7770                     [1604]
130 MEMORY 34611                            [F058]
140 LOAD "bp.bin",34612                     [073C]
160 INK 0,0:INK 5,1:INK 14,15:INK 15,9      [7A9A]
170 bladr=&C000:adro=bladr                  [3C2A]
175 ovly=&8734:mirry=&875B:sr=&878C:sl=&
    87B2:menu=&87D8:picm=&880F:abfall=&8
    B00:compare=&8B1B:cfl=&8B62:copy=&8B
    63:lbuff=&8B83:prtable=&8BCD:mochr=&
    8BF8:chpic=&8C0B:clbyte=&8BA0           [5A8C]
180 zeich=&8C2A:vzeich=&8C3F:curcl=&8C4C
    :curset=&8C6B:feuer=&8C9A:clscreen=&
    8CCA:prscreen=&8CDB:zoom=&8D03:vcur=
    &8DD6:vfeuer=&8E1D:vpoint=&8E2A:deco
    de=&8E45:encode=&8E4F:adrcl=&8E73:xp
    o=&8E76:ypo=&8E77                       [EA4E]
190 spx=&8E71:spy=&8E72:xcc=&8E74:ycc=&8
    E75:adrchr=&967B:buffer=&8E8B:adrcol
    =&9283:picture=&9293:chrnum=&8E79       [0574]
195 temp$=STRING$(16,CHR$(0)):tempadr=PE
    EK(@temp$+1)+256*PEEK(@temp$+2)         [00CC]
200 my=1:xc=0:yc=0:color=1:newchr=1:actc
    hr=0:y=0                                [2B7A]
201 DIM co(7)                               [980E]
202 FOR i=0 TO 7:READ a:co(i)=a:NEXT        [DBBE]
203 DATA 0,24,14,6,26,10,18,8               [477C]
205 MODE 0:BORDER 0:POKE adrchr,0           [A09A]
206 FOR i=0 TO 15:INK i,0:NEXT:GOSUB 100
    00                                      [D746]
220 '                                       [06B6]
225 CALL clscreen:CALL prscreen             [A938]
230 adro=bladr                              [8D3A]
240 POKE chrnum,actchr                      [3366]
245 madrn=bladr+4+my*80:madr=madrn          [017C]
250 '                                       [E1BC]
260 '   ###### m e n u ######               [8D30]
270 '                                       [E3C0]
280 BORDER 0:GOSUB 490:CALL menu,picm+y:
    CALL curset,madr                        [597E]
300 in$=INKEY$:IF in$="" THEN 300           [79EC]
302 IF ASC(in$)=88 OR ASC(in$)=224 THEN
    306                                     [D42A]
305 GOTO 410                                [064C]
306 IF INT((my-1+y/5)/2)<>(my-1+y/5)/2 T
    HEN 300                                 [A938]
307 ON (my-1+y/5)/2+1 GOTO 500,3200,6000
    ,3900,1645,345,346,6200,7000,2800,23
    60,6300,3600,7140                       [2DFA]
345 mflag=0:BORDER 8:GOTO 2500              [AAB4]
346 mflag=1:BORDER 7:GOTO 2500              [ACB6]
410 IF ASC(in$)=10 OR ASC(in$)=241 THEN
    IF my=23 THEN IF y=20 THEN 300 ELSE
    y=y+5:GOTO 450 ELSE my=my+1:GOTO 440
                                            [31F0]
420 IF ASC(in$)=11 OR ASC(in$)=240 THEN
    IF my=1 THEN IF y=0 THEN 300 ELSE y=
    y-5:GOTO 450 ELSE my=my-1:GOTO 440      [512E]
430 GOTO 300                                [E546]
440 CALL curcl,madr:madr=bladr+4+my*80:C
    ALL curset,madr:GOTO 300                [A102]
450 CALL menu,picm+y:CALL curset,madr:GO
    TO 300                                  [644A]
480 FOR i=0 TO 15:INK i,PEEK(adrcol+i):N
    EXT:RETURN                              [BA8A]
490 FOR i=0 TO 7:INK i,co(i):NEXT:RETURN
                                            [6848]
500 '                                       [06B8]
510 '   ###### p a i n t ######             [AB3A]
520 '                                       [08BC]
525 BORDER 1:GOSUB 480:adro=bladr+xc*2+y
    c*80:CALL prscreen:CALL curset,adro     [6428]
530 in$=INKEY$:IF in$="" THEN 530           [6C00]
540 kfl=0:GOSUB 1150:IF kfl THEN GOSUB 1
    221:GOTO 530                            [F854]
545 IF ASC(in$)=88 OR ASC(in$)=224 THEN
    CALL feuer,adro:CALL curset,adro:POK
    E picture+xc+40*yc,PEEK(chrnum):GOTO
     640                                    [1508]
546 IF in$=" " THEN GOSUB 660:GOTO 640      [FDC4]
560 IF ASC(in$)=4 THEN ndfl=1:GOTO 700      [3252]
565 IF ASC(in$)=18 THEN ndfl=0:GOTO 700     [6FC4]
570 IF ASC(in$)=7 THEN IF PEEK(picture+x
    c+40*yc)=0 THEN 530 ELSE actchr=PEEK
    (picture+xc+40*yc):POKE chrnum,actch
    r:GOSUB 7780:GOTO 530                   [00B6]
590 IF ASC(in$)=20 THEN GOSUB 1390:GOTO
    530                                     [116C]
600 IF ASC(in$)=13 THEN CALL curcl,adro:
    GOTO 280                                [11D8]
605 IF ASC(in$)=26 THEN 1650                [49DA]
606 IF ASC(in$)=14 THEN LOCATE 1,1:PEN 1
    :PRINT"chr.code :";PEEK(picture+xc+4
    0*yc):FOR w=0 TO 1000:NEXT:CALL prsc
    reen:CALL curset,adro                   [703C]
630 GOTO 530                                [D454]
640 IF xc=39 THEN 530 ELSE xc=xc+1:GOSUB
     1221:GOTO 530                          [C5EE]
660 POKE chrnum,0:CALL feuer,adro:CALL c
    urset,adro:POKE picture+xc+40*yc,0:P
    OKE chrnum,actchr:RETURN                [24FC]
670 '                                       [B3C8]
680 ' ****** zeichen definieren ****** (
    auf normalen bildschirm)                [848C]
690 '                                       [E5CC]
700 IF newchr=256 THEN GOSUB 9030:CALL p
    rscreen:CALL curset,adro:GOTO 500       [0F00]
710 CALL curcl,adro:rfl=0:GOSUB 1080:IF
    rfl THEN CALL curset,adro:GOTO 530      [C2B0]
720 xpoia=xc*16:ypoia=(24-yc)*16+15         [5AFC]
730 '                                       [E7C2]
740 GOSUB 920                               [EEEC]
750 IF TEST(xpoia+xpoir,ypoia-ypoir) THE
    N col=0 ELSE col=color                  [408A]
760 PLOT xpoia+xpoir,ypoia-ypoir,col        [1D64]
770 '                                       [E3CA]
780 in$=INKEY$:IF in$="" THEN 780           [6A1C]
790 IF ASC(in$)=2 THEN POKE chrnum,PEEK(
    picture+xc+40*yc):CALL feuer,adro:PO
    KE chrnum,actchr:CALL curset,adro:GO
    TO 530                                  [7148]
800 IF ASC(in$)=3 THEN GOSUB 1260:GOTO 7
    80                                      [D808]
810 IF ASC(in$)=13 THEN 960                 [7574]
820 IF ASC(in$)=26 THEN GOSUB 1850:GOSUB
     1790:pladr=&C2A0+xpoir+ypoir*40:CAL
    L zoom,adrzoom:CALL vzeich,tempadr:f
    l=0:GOTO 1930                           [59D8]
830 IF ASC(in$)=16 THEN POKE chrnum,0:CA
    LL feuer,adro:FOR i=0 TO 15:POKE tem
    padr+i,0:NEXT:PLOT xpoia+xpoir,ypoia
    -ypoir,color:GOTO 780                   [2E70]
840 keyfl=0:GOSUB 1020:IF keyfl THEN GOS
    UB 880:GOTO 740                         [BE84]
850 IF ASC(in$)=88 OR ASC(in$)=224 THEN
    GOSUB 920:GOSUB 865:GOSUB 880:GOTO 7
    40                                      [13C8]
860 GOTO 780                                [F36C]
865 IF xpoir=12 THEN xpoir=0:RETURN ELSE
     xpoir=xpoir+4:RETURN                   [26DE]
870 '                                       [B3CC]
880 POKE adro+oldy*1024,PEEK(tempadr+old
    y/2)                                    [F2B8]
890 POKE adro+oldy*1024+1,PEEK(tempadr+o
    ldy/2+8)                                [7338]
900 oldy=ypoir:RETURN                       [ADF6]
910 '                                       [0DC2]
920 POKE tempadr+ypoir/2,PEEK(adro+ypoir
    *1024)                                  [259A]
930 POKE tempadr+ypoir/2+8,PEEK(adro+ypo
    ir*1024+1)                              [011A]
940 RETURN                                  [A23A]
950 '                                       [11CA]
960 ' zeichen uebernehmen                   [F234]
965 POKE &8B44,8:POKE &8B4D,&23             [6EAE]
970 CALL compare,newchr,tempadr:IF PEEK(
    cfl)<>0 THEN IF (newchr-PEEK(cfl)=ch
    r AND ndfl=0) THEN 975 ELSE chr=newc
    hr-PEEK(cfl):POKE picture+xc+yc*40,c
    hr:GOSUB 9050:ndfl=0:GOTO 980           [86BE]
975 FOR i=0 TO 15:POKE chrzei+i,PEEK(tem
    padr+i):NEXT                            [CC52]
980 actchr=chr:POKE chrnum,chr:CALL feue
    r,adro                                  [571C]
990 IF ndfl THEN POKE picture+xc+yc*40,c
    hr:newchr=newchr+1 ELSE CALL prscree
    n                                       [7B76]
1000 CALL curset,adro:GOTO 530              [AC60]
1010 '                                      [9212]
1020 ' punkt bei der definition bewegen     [349A]
1030 IF ASC(in$)=9 OR ASC(in$)=243 THEN
     keyfl=1:IF xpoir=12 THEN xpoir=0 EL
     SE xpoir=xpoir+4                       [9860]
1040 IF ASC(in$)=8 OR ASC(in$)=242 THEN
     keyfl=1:IF xpoir=0 THEN xpoir=12 EL
     SE xpoir=xpoir-4                       [7F62]
1050 IF ASC(in$)=10 OR ASC(in$)=241 THEN
      keyfl=1:IF ypoir=14 THEN ypoir=0 E
     LSE ypoir=ypoir+2                      [69B8]
1060 IF ASC(in$)=11 OR ASC(in$)=240 THEN
      keyfl=1:IF ypoir=0 THEN ypoir=14 E
     LSE ypoir=ypoir-2                      [70BE]
1070 RETURN                                 [AB90]
1080 chr=PEEK(picture+xc+40*yc)             [1AE0]
1082 chrzei=adrchr+chr*16                   [3334]
1084 FOR i=0 TO 7:POKE tempadr+i,PEEK(ch
     rzei+i):POKE tempadr+i+8,PEEK(chrze
     i+i+8):NEXT                            [325A]
1090 IF ndfl THEN chr=newchr ELSE IF chr
     =0 THEN rfl=1:RETURN                   [8E54]
1110 adro=bladr+xc*2+yc*80                  [758C]
```

Listing 1. Steuerprogramm für »Background Painter«

```
1120 chrzei=adrchr+chr*16                    [F026]
1140 xpoir=0:ypoir=0:oldy=0:RETURN           [CFB0]
1150 '                                       [921C]
1160 '  ****** bewege cursor ******          [0768]
1170 '                                       [B420]
1180 IF ASC(in$)=9 OR ASC(in$)=243 THEN
     IF xc=39 THEN 1220 ELSE xc=xc+1:kfl
     =1:RETURN                               [1732]
1190 IF ASC(in$)=8 OR ASC(in$)=242 THEN
     IF xc=0 THEN 1220 ELSE xc=xc-1:kfl=
     1:RETURN                                [7CBC]
1200 IF ASC(in$)=10 OR ASC(in$)=241 THEN
      IF yc=24 THEN 1220 ELSE yc=yc+1:kf
     l=1:RETURN                              [C86A]
1210 IF ASC(in$)=11 OR ASC(in$)=240 THEN
      IF yc=0 THEN 1220 ELSE yc=yc-1:kfl
     =1:RETURN                               [3104]
1220 RETURN                                  [AB8A]
1221 CALL curcl,adro:adro=bladr+xc*2+yc*
     80                                      [9054]
1222 CALL curset,adro:RETURN                 [774A]
1230 '                                       [B81A]
1240 '  ****** farbe fuer punkt auswaehl
     en ******                               [7EEE]
1250 '                                       [921E]
1260 BORDER PEEK(adrcol+color):fl1=0         [7A4C]
1270 GOSUB 1300:IF col THEN PLOT xpoia+x
     poir,ypoia-ypoir,color                  [5374]
1280 FOR w=1 TO 100:NEXT:IF fl1 THEN RET
     URN ELSE GOTO 1270                      [CA16]
1290 '                                       [DE26]
1300 in$=INKEY$:IF in$="" THEN 1300          [F7B0]
1310 IF ASC(in$)=9 OR ASC(in$)=243 THEN
     IF color=15 THEN 1300 ELSE color=co
     lor+1:GOTO 1350                         [A86C]
1320 IF ASC(in$)=8 OR ASC(in$)=242 THEN
     IF color=1 THEN 1300 ELSE color=col
     or-1:GOTO 1350                          [5A04]
1330 IF ASC(in$)=88 OR ASC(in$)=224 THEN
      BORDER 1:fl1=1:RETURN                  [3704]
1340 GOTO 1300                               [920A]
1350 BORDER PEEK(adrcol+color):RETURN        [392C]
1360 '                                       [B522]
1370 '  ###### c h a r a c t e r   t a b
     l e ######                              [1956]
1380 '                                       [BB26]
1390 PAPER #1,0:GOSUB 1410:GOTO 1560         [DF68]
1400 '                                       [B918]
1410 CALL curcl,adro:GOSUB 1420:GOTO 151
     0                                       [3E10]
1420 WINDOW #1,1,20,1,INT((newchr-1)/20)
     +4:PAPER #1,0:CLS #1                    [0436]
1430 PEN 2:LOCATE 1,1:PRINT "{CTRL X}cha
     racter table{CTRL X}{CTRL O}{CTRL C
     }";0;                                   [544A]
1440 PEN 1:PRINT STRING$(20,"-");            [4914]
1450 LOCATE 1,INT((newchr-1)/20)+4:PRINT
      STRING$(20,"-"):PEN 3                  [2746]
1460 count=0:xc1=0:adro1=bladr+80*2          [82EC]
1470 IF newchr>1 THEN CALL prtable,newch
     r                                       [56B2]
1480 POKE chrnum,actchr                      [39D4]
1490 CALL curset,adro1:RETURN                [C7BA]
1500 '                                       [B71A]
1510 in$=INKEY$:IF in$="" THEN 1510          [E0BC]
1520 IF ASC(in$)=9 OR ASC(in$)=243 THEN
     IF count=newchr-1 THEN xc1=0:count=
     0:GOTO 1590 ELSE xc1=xc1+2:count=co
     unt+1:GOTO 1590                         [825E]
1530 IF ASC(in$)=8 OR ASC(in$)=242 THEN
     IF count=0 THEN count=newchr-1:xc1=
     (newchr-1)*2:GOTO 1590 ELSE xc1=xc1
     -2:count=count-1:GOTO 1590              [2E28]
1540 IF ASC(in$)=88 OR ASC(in$)=224 THEN
      RETURN                                 [EB22]
1550 GOTO 1510                               [BC16]
1560 IF count<>0 THEN actchr=count:POKE
     chrnum,actchr                           [98DE]
1570 CALL prscreen:CALL curset,adro:RETU
     RN                                      [A406]
1580 '                                       [DF2A]
1590 LOCATE 16,1:PRINT count;" ":adrn1=b
     ladr+xc1*2+160                          [C9C2]
1600 CALL curcl,adro1:CALL curset,adrn1      [D39A]
1610 adro1=adrn1:GOTO 1510                   [FC58]
1620 '                                       [BF20]
1630 '  ###### z o o m ######                [DDB4]
1640 '                                       [C124]
1645 BORDER 1:GOSUB 480                      [88EA]
1650 GOSUB 1850                              [9CB4]
1660 GOSUB 1790                              [B4BC]
1670 PEN 3:LOCATE 8,25:PRINT xc;:LOCATE
     17,25:PRINT yc;:LOCATE 1,1              [8152]
1680 CALL zoom,adrzoom:CALL vcur,adrvcur
                                             [18E2]
1690 '                                       [962E]
1700 in$=INKEY$:IF in$="" THEN 1700          [D3C0]
1705 kfl=0:GOSUB 1150:IF kfl THEN 1660       [B234]
1706 IF ASC(in$)=88 OR ASC(in$)=224 THEN
      CALL vfeuer:POKE picture+xc+40*yc.
     PEEK(chrnum):GOTO 1775                  [C268]
1710 IF ASC(in$)=26 THEN CALL prscreen:a
     dro=bladr+xc*2+yc*80:CALL curset,ad
     ro:GOTO 530                             [A926]
1720 IF in$=" " THEN POKE chrnum,0:CALL
     vfeuer:POKE picture+xc+40*yc,0:POKE
      chrnum,actchr:GOTO 1775                [DFF2]
1730 IF ASC(in$)=7 THEN IF PEEK(picture+
     xc+40*yc)=0 OR PEEK(chrnum)=0 THEN
     1700 ELSE actchr=PEEK(picture+xc+40
     *yc):POKE chrnum,actchr:GOSUB 1790:
     CALL zoom,adrzoom:FOR w=1 TO 100:NE
     XT:CALL vcur,adrvcur                    [8A54]
1740 IF ASC(in$)=4 THEN ndfl=1:GOTO 1900
                                             [1C1A]
1745 IF ASC(in$)=18 THEN ndfl=0:GOTO 190
     0                                       [5F8C]
1750 IF ASC(in$)=13 THEN 280                 [A1D2]
1770 GOTO 1700                               [C920]
1775 IF xc=39 THEN 1700 ELSE xc=xc+1:GOT
     O 1660                                  [DFEC]
1780 '                                       [232E]
1790 adrzoom=picture+xc-2+40*(yc-1)          [F09E]
1800 adrvcur=PEEK(picture+xc+40*yc)*16+a
     drchr                                   [FFB4]
1810 IF xc<=1 THEN xp=254+xc ELSE xp=xc-
     2                                       [F136]
1820 IF yc=0 THEN yp=255 ELSE yp=yc-1        [80BA]
1830 POKE xpo,xp:POKE ypo,yp:RETURN          [A16C]
1840 '                                       [E528]
1850 PEN 2:LOCATE 1,25:PRINT "{2 SPACE}x
     pos{6 SPACE}ypos{4 SPACE}";:LOCATE
     1,1                                     [1728]
1860 PEN 3:LOCATE 8,25:PRINT xc;:LOCATE
     17,25:PRINT yc;:LOCATE 1,1:RETURN       [D788]
1870 '                                       [E42E]
1880 '  ****** zeichen definieren ******
      (auf vergroessertem bildschirm)        [FD74]
1890 '                                       [9A32]
1900 IF newchr=256 THEN GOSUB 9030:GOTO
     1680                                    [666E]
1910 rfl=0:GOSUB 1080:IF rfl THEN 1660 E
     LSE pladr=&C2A0:POKE xpo,xp:POKE yp
     o,yp                                    [9AAE]
1920 fl=0:CALL zoom,adrzoom                  [2A1C]
1930 POKE adrcl,color:tempcl=PEEK(pladr)
     :CALL vpoint,pladr                      [5CCE]
1940 '                                       [E32A]
1950 in$=INKEY$:IF in$="" THEN 1950          [90DC]
1953 keyfl=0:GOSUB 1020:IF keyfl THEN 20
     50                                      [ADA2]
1954 IF ASC(in$)=88 OR ASC(in$)=224 THEN
      GOSUB 2080:GOTO 2320                   [57FC]
1960 IF ASC(in$)=2 THEN POKE chrnum,PEEK
     (picture+xc+40*yc):CALL vfeuer:CALL
      vcur,PEEK(chrnum)*16+adrchr:POKE c
     hrnum,actchr:GOTO 1700                  [B2BE]
1970 IF ASC(in$)=3 THEN 2290                 [2EDE]
1980 IF ASC(in$)=13 THEN 2220                [8934]
1990 IF ASC(in$)=26 THEN CALL prscreen:C
     ALL zeich,adro,tempadr:oldy=ypoir:G
     OTO 720                                 [3EF2]
1995 IF ASC(in$)=32 THEN IF PEEK(pladr)=
     0 AND tempcl=0 THEN 1950 ELSE adr=t
     empadr+ypoir/2-(xpoir>=8)*8:GOSUB 2
     170:GOTO 2320                           [D21E]
2000 IF ASC(in$)=16 THEN POKE chrnum,0:C
     ALL vfeuer:FOR i=0 TO 15:POKE tempa
     dr+i,0:NEXT:CALL vpoint,pladr:GOTO
     1950                                    [DEF0]
2005 IF ASC(in$)=19 THEN 2251                [3432]
2010 IF ASC(in$)=244 THEN GOSUB 2313:CAL
     L vzeich,tempadr:tempcl=PEEK(pladr)
     :CALL vpoint,pladr:GOTO 1950            [F570]
2015 IF ASC(in$)=245 THEN GOSUB 2314:CAL
     L vzeich,tempadr:tempcl=PEEK(pladr)
     :CALL vpoint,pladr:GOTO 1950            [107E]
2020 IF ASC(in$)=247 THEN CALL sr,tempad
     r:CALL vzeich,tempadr:tempcl=PEEK(p
     ladr):CALL vpoint,pladr:GOTO 1950       [7D1A]
2025 IF ASC(in$)=246 THEN CALL sl,tempad
     r:CALL vzeich,tempadr:tempcl=PEEK(p
     ladr):CALL vpoint,pladr:GOTO 1950       [F316]
2030 IF ASC(in$)=248 OR ASC(in$)=249 THE
     N GOSUB 2315:CALL vzeich,tempadr:te
     mpcl=PEEK(pladr):CALL vpoint,pladr:
     GOTO 1950                               [6140]
2035 IF ASC(in$)=250 OR ASC(in$)=251 THE
     N CALL mirry,tempadr:CALL vzeich,te
     mpadr:tempcl=PEEK(pladr):CALL vpoin
     t,pladr:GOTO 1950                       [0168]
2036 IF ASC(in$)=15 THEN 2040                [C42A]
2039 GOTO 1950                               [852C]
2040 PAPER #1,0:GOSUB 1420:GOSUB 1510:IF
      count=0 THEN 2048                      [1858]
2041 CALL ovly,tempadr,adrchr+16*count       [7716]
```

```
2048 GOSUB 1790:CALL zoom,adrzoom:CALL v
     zeich,tempadr:GOTO 1930              [BD74]
2050 POKE adrcl,tempcl:CALL decode:CALL
     vpoint,pladr:IF PEEK(adrcl)<>0 AND
     fl THEN CALL vpoint,pladr:fl=0       [9A26]
2060 pladr=&C2A0+xpoir+ypoir*40:GOTO 193
     0                                    [CCDA]
2070 '                                    [9720]
2080 IF PEEK(pladr)=0 AND tempcl=0 THEN
     1950                                 [60A4]
2090 adr=tempadr+ypoir/2-(xpoir>=8)*8     [B4A0]
2100 fl=1:IF tempcl<>0 THEN GOSUB 2170    [E828]
2110 POKE adrcl,color:CALL encode         [89D0]
2120 code=PEEK(adrcl):tempcl=code+code*2
                                          [27E4]
2130 IF xpoir=0 OR xpoir=8 THEN code=cod
     e*2                                  [56F4]
2140 POKE adr,PEEK(adr)+code              [3D18]
2150 RETURN                               [8E90]
2160 '                                    [9620]
2170 code=PEEK(adr)                       [EADE]
2180 POKE adrcl,tempcl:CALL decode:CALL
     encode                               [3D5E]
2190 IF xpoir=0 OR xpoir=8 THEN code=cod
     e-PEEK(adrcl)*2 ELSE code=code-PEEK
     (adrcl)                              [2B5C]
2200 POKE adr,code:tempcl=0:fl=0:RETURN   [7DE0]
2210 ' zeichen uebernehmen                [6680]
2220 POKE &8B44,8:POKE &8B4D,&23          [0DF2]
2225 CALL compare,newchr,tempadr:IF PEEK
     (cfl)<>0 THEN IF (newchr-PEEK(cfl)=
     chr AND ndfl=0) THEN 2228 ELSE chr=
     newchr-PEEK(cfl):POKE picture+xc+yc
     *40,chr:GOSUB 9050:ndfl=0:GOTO 2230
                                          [A2B2]
2228 FOR i=0 TO 15:POKE chrzei+i,PEEK(te
     mpadr+i):NEXT                        [A8A4]
2230 actchr=chr:POKE chrnum,chr:CALL vfe
     uer                                  [BDB0]
2240 IF ndfl THEN POKE picture+xc+yc*40,
     chr:newchr=newchr+1 ELSE POKE xpo,x
     p:POKE ypo,yp:CALL zoom,adrzoom      [B6AC]
2250 CALL vcur,chr*16+adrchr:GOTO 1700    [7AB2]
2251 ' suche zeichen (ctrl s)             [1A10]
2252 POKE &8B44,8:POKE &8B4D,&23          [9AFC]
2253 CALL compare,newchr,tempadr:IF PEEK
     (cfl)<>0 THEN GOSUB 9130 ELSE GOSUB
      9140                                [A098]
2254 GOSUB 1850:GOSUB 1790:CALL zoom,adr
     zoom:CALL vzeich,tempadr:fl=0:GOTO
     1930                                 [E8B4]
2260 '                                    [9822]
2270 ' ****** farbe fuer punkt auswaehl
     en ****** (bei vergr. blildschirm)   [217A]
2280 '                                    [2626]
2290 BORDER PEEK(adrcol+color):fl1=0      [2454]
2300 GOSUB 1300                           [5292]
2310 POKE adrcl,color:CALL vpoint,pladr:
     CALL vpoint,pladr                    [9260]
2311 FOR w=1 TO 100:NEXT:IF fl1 THEN 195
     0 ELSE GOTO 2300                     [FDE2]
2312 '                                    [C11E]
2313 FOR j=0 TO 6:POKE tempadr+j,PEEK(te
     mpadr+j+1):POKE tempadr+j+8,PEEK(te
     mpadr+j+9):NEXT:POKE tempadr+7,0:PO
     KE tempadr+15,0:RETURN               [2216]
2314 FOR j=7 TO 1 STEP -1:POKE tempadr+j
     ,PEEK(tempadr+j-1):POKE tempadr+j+8
     ,PEEK(tempadr+j+7):NEXT:POKE tempad
     r,0:POKE tempadr+8,0:RETURN          [44B0]
2315 FOR j=0 TO 3:temp=PEEK(tempadr+j):P
     OKE tempadr+j,PEEK(tempadr+7-j):POK
     E tempadr+7-j,temp:temp=PEEK(tempad
     r+j+8):POKE tempadr+j+8,PEEK(tempad
     r+15-j):POKE tempadr+15-j,temp:NEXT
     :RETURN                              [6150]
2320 IF xpoir=12 THEN xpoir=0:GOTO 2050
     ELSE xpoir=xpoir+4:GOTO 2050         [1926]
2330 '                                    [C31E]
2340 '  ###### clear character ######     [6A10]
2350 '                                    [E522]
2360 CALL curcl,madr:PAPER #1,0:GOSUB 48
     0                                    [3910]
2370 WINDOW #1,1,20,1,INT((newchr-1)/20)
     +4:CLS #1                            [7C3C]
2380 LOCATE 1,1:PEN 3:PRINT"{CTRL X}clea
     r character{CTRL X}":0               [4E7A]
2390 temp=actchr:GOSUB 1440:GOSUB 1510    [9434]
2400 IF count=0 THEN CALL prscreen:actch
     r=temp:POKE chrnum,temp:GOSUB 490:I
     F newchr=1 THEN CALL clscreen:GOTO
     280 ELSE GOTO 280                    [7654]
2410 IF temp>=count THEN IF temp=count T
     HEN temp=0 ELSE temp=temp-1          [8558]
2420 IF count<> newchr-1 THEN CALL mochr
     ,(newchr-1-count)*16,count*16+adrch
     r ELSE CALL chpic,count:count=count
     -1:xc1=xc1-2:adro1=bladr+xc1*2+160:
     GOTO 2440                            [D0D8]
2430 CALL chpic,count                     [ED02]
2440 newchr=newchr-1:GOSUB 1470:GOSUB 15
     10:GOTO 2400                         [F166]
2445 '                                    [962C]
2450 FOR i=1 TO 25:LOCATE 1,26:PRINT:NEX
     T:MODE 0:CALL prscreen:CALL curset,
     adro:RETURN                          [4252]
2500 '                                    [941C]
2510 '  ###### c o p y  and  m o v e ###
     ###                                  [1C30]
2520 '                                    [9220]
2525 fl=0:rfl=0:GOSUB 2530:IF rfl THEN C
     ALL prscreen:GOTO 280 ELSE CALL cop
     y,picture+tx+40*ty:GOTO 2600         [9F86]
2530 GOSUB 480:CALL prscreen:PEN 1:LOCAT
     E 1,1:PRINT"{CTRL X}{5 SPACE}set ma
     rks{6 SPACE}{CTRL X}":FOR w=1 TO 10
     00:NEXT:CALL prscreen:CALL curset,a
     dro                                  [9B54]
2540 GOSUB 2640:IF spfl THEN 2530 ELSE I
     F rfl THEN RETURN ELSE GOSUB 2700    [A256]
2550 tx=xc:ty=yc                          [0E60]
2560 GOSUB 2640:IF spfl THEN 2530 ELSE I
     F rfl THEN RETURN ELSE GOSUB 2700    [965A]
2570 IF xc<tx OR yc<ty THEN GOSUB 9040:G
     OTO 2530                             [617E]
2580 POKE spx,xc-tx+1:POKE spy,yc-ty+1    [F6F0]
2590 RETURN                               [BAA0]
2600 GOSUB 2640:IF spfl THEN 2525 ELSE I
     F rfl THEN CALL prscreen:GOTO 280 E
     LSE IF mflag THEN POKE xcc,tx:POKE
     ycc,ty:POKE clbyte,&3E:CALL lbuff,p
     icture+tx+40*ty:POKE clbyte,&1A      [24D2]
2602 POKE xcc,xc:POKE ycc,yc:CALL lbuff,
     picture+xc+40*yc                     [452C]
2605 CALL prscreen:CALL curset,adro       [30D2]
2610 tx=xc:ty=yc:GOTO 2600                [FC10]
2640 in$=INKEY$:IF in$="" THEN 2640       [C7D0]
2645 spfl=0:IF in$=" " THEN spfl=1:RETUR
     N                                    [4988]
2647 IF ASC(in$)=13 THEN CALL curcl,adro
     :rfl=1:RETURN                        [6B04]
2650 kfl=0:GOSUB 1150:IF kfl THEN GOSUB
     1221:GOTO 2640                       [BA24]
2660 IF ASC(in$)=88 OR ASC(in$)=224 THEN
      RETURN                              [4C2A]
2670 GOTO 2640                            [9328]
2700 CALL curcl,adro:FOR w=1 TO 100:NEXT
     :CALL curset,adro                    [1D02]
2710 FOR i=0 TO 7:POKE buffer+i,PEEK(adr
     o+i*2048):POKE buffer+i+8,PEEK(adro
     +1+i*2048):NEXT i:RETURN             [6B24]
2800 '                                    [EA22]
2810 '  ###### c l e a r  s c r e e n ##
     ####                                 [843A]
2820 '                                    [2C26]
2830 PEN 3:LOCATE 1,1:PRINT"{CTRL X}{4 S
     PACE}clear screen{4 SPACE}{CTRL X}"
     ;:PEN 1:PRINT STRING$(20,"-")        [8800]
2840 WINDOW #1,4,17,4,10:PAPER #1,7:CLS
     #1                                   [5A68]
2850 LOCATE #1,1,2:PEN #1,3:PRINT#1," {C
     TRL X}a{CTRL X}ll or {CTRL X}p{CTRL
      X}art"                              [E7C4]
2860 in$=INKEY$:IF in$="a" THEN 2870 ELS
     E IF in$="p" THEN 2900 ELSE GOTO 28
     60                                   [BB44]
2870 GOSUB 3000:CALL clscreen:CLS:adro=b
     ladr:xc=0:yc=0:GOTO 525              [7B16]
2900 GOSUB 3000                           [EA9C]
2910 rfl=0:GOSUB 2530:IF rfl THEN 280 EL
     SE POKE clbyte,&3E:POKE xcc,tx:POKE
      ycc,ty:CALL lbuff,picture+tx+40*ty
     :POKE clbyte,&1A:GOTO 2910           [E8AA]
3000 PEN #1,0:PRINT#1:PRINT#1," are you
     sure":PRINT#1,"{5 SPACE}y/n"         [6262]
3010 in$=INKEY$:IF in$="y" THEN RETURN E
     LSE IF in$="n" THEN CALL prscreen:G
     OTO 280 ELSE GOTO 3010               [AB5E]
3200 '                                    [9318]
3210 '  ###### c o l o r  m e n u  ####
     ##                                   [E2CA]
3220 '                                    [911C]
3230 GOSUB 480                            [1048]
3231 CLS:PEN 2:LOCATE 1,1:PRINT "{CTRL X
     }{4 SPACE}color menu{6 SPACE}{CTRL
     X}";:PEN 1:PRINT STRING$(20,"-")     [4088]
3235 WINDOW #1,1,20,3,12:PAPER #1,0:CLS
     #1                                   [3348]
3240 FOR i=0 TO 7                         [E9C0]
3250    LOCATE 2,4+i:PEN 1:PRINT i;:PEN
     i:PRINT"{CTRL X}{5 SPACE}{CTRL X}";
     :PEN i+8:PRINT "{CTRL X}{5 SPACE}{C
     TRL X}";:PEN 1:PRINT i+8             [73BA]
3260 NEXT:PRINT"{CTRL O}{CTRL B}ctrl a:"
     :PRINT "{CTRL O}{CTRL C}{2 SPACE}ch
     ange on picture":PEN 1:PRINT:PRINT
     STRING$(20,"-");                     [92A8]
3270 PRINT:PRINT"{CTRL O}{CTRL B}{2 SPAC
```

```
     E}ink":PRINT:PRINT"{2 SPACE}color"   [ED90]
3280 fl=0:col1=0:ink1=0:xc=8:yc=3:adro=b
     ladr+xc*2+80*yc:CALL curset,adro     [E1CE]
3290 GOSUB 3580                            [3DBC]
3300 in$=INKEY$:IF in$="" THEN 3300        [B5B8]
3310 IF ASC(in$)=13 THEN CALL prscreen:G
     OTO 280                               [4028]
3400 kfl=0:GOSUB 1150:IF kfl THEN GOSUB
     1221                                  [405A]
3410 IF xc<8 OR xc>27 OR yc<3 OR yc>10 T
     HEN 3300                              [DB0C]
3415 GOSUB 3580                            [B2BA]
3420 IF ASC(in$)=88 OR ASC(in$)=224 THEN
      3450                                 [D4F8]
3430 GOTO 3300                             [C312]
3450 CALL curcl,adro                       [9226]
3500 in$=INKEY$:IF in$="" THEN 3500        [C7C0]
3510 IF ASC(in$)=9 OR ASC(in$)=243 THEN
     IF col1=26 THEN 3500 ELSE col1=col1
     +1:GOTO 3550                          [0768]
3520 IF ASC(in$)=8 OR ASC(in$)=242 THEN
     IF col1=0 THEN 3500 ELSE col1=col1-
     1:GOTO 3550                           [E0FA]
3530 IF ASC(in$)=88 OR ASC(in$)=224 THEN
      POKE adrcol+ink1,col1:IF fl THEN 3
     230 ELSE CALL curset,adro:GOTO 3300
                                           [970E]
3535 IF ASC(in$)=1 THEN fl=1:CALL prscre
     en                                    [176E]
3540 GOTO 3500                             [721A]
3550 INK ink1,col1                         [8DFA]
3560 IF fl=0 THEN GOSUB 3590               [D078]
3570 GOTO 3500                             [CD20]
3580 ink1=yc-3-(xc>17)*8:col1=PEEK(adrco
     l+ink1)                               [7E58]
3590 IF PEEK(adrcol)=col1 OR PEEK(adrcol
     +3)=col1 THEN PEN 2 ELSE PEN 3        [56A0]
3595 LOCATE 9,17:PRINT ink1:LOCATE 9,19:
     PRINT col1:RETURN                     [578C]
3600 '                                     [BB20]
3610 '  ###### v a l u e   t a b l e ####
     ##                                    [3934]
3620 '                                     [9924]
3630 MODE 2:INK 1,24:fl=0:flag=0           [C638]
3640 PRINT STRING$(80,"-");"{16 SPACE}##
     #### V A L U E{2 SPACE}T A B L E ##
     ####":PRINT STRING$(80,"-"):PRINT:W
     INDOW #0,1,80,6,25                    [9FBA]
3641 PRINT:PRINT"{CTRL X}s{CTRL X}creen
     or {CTRL X}p{CTRL X}rinter :":PRINT
                                           [7658]
3642 in$=INKEY$:IF in$="s" THEN n=0 ELSE
      IF in$="p" THEN n=8 ELSE GOTO 3642
                                           [16A8]
3645 PRINT #n,"values of character set :
     ":PRINT #n                            [A860]
3647 zeig=adrchr                           [35A8]
3648 PRINT"{CTRL X}a{CTRL X}ll or {CTRL
     X}s{CTRL X}equentiell :"              [7B0A]
3649 in$=INKEY$:IF in$="a" THEN 3650 ELS
     E IF in$="s" THEN 3655 ELSE GOTO 36
     49                                    [856C]
3650 FOR i=0 TO newchr-1                   [DD26]
3651   GOSUB 3660                          [4C3C]
3652 NEXT:GOSUB 3810                       [49AA]
3654 GOSUB 3770:IF flag THEN 3820 ELSE G
     OTO 3654                              [046C]
3655 PRINT:INPUT"enter code of character
     :";i:IF i>=newchr OR i<0 THEN PRINT
     :PRINT"not defined !!!":GOTO 3655     [5482]
3656 GOSUB 3660:PRINT:PRINT"press {CTRL
     X}c{CTRL X} to continue or {CTRL X}
     enter{CTRL X} to return "             [83DE]
3657 in$=INKEY$:IF in$="" THEN 3657 ELSE
      IF ASC(in$)=13 THEN 3820 ELSE IF i
     n$="c" THEN 3655 ELSE GOTO 3657       [9BC4]
3660 PRINT #n,"char.:";i                   [498E]
3670 PRINT #n,"{4 SPACE}l.s.: &";HEX$(PE
     EK(zeig+i*16));                       [70E6]
3680 FOR j=1 TO 7                          [57D4]
3690   PRINT #n,",&";HEX$(PEEK(zeig+j+i*
     16));                                 [A4C2]
3700 NEXT                                  [F052]
3710 PRINT #n:PRINT #n,"{4 SPACE}r.s.: &
     ";HEX$(PEEK(zeig+j+i*16));            [4E02]
3720 FOR j=9 TO 15                         [6138]
3730   PRINT #n,",&";HEX$(PEEK(zeig+j+i*
     16));                                 [D3B8]
3740 NEXT:PRINT #n                         [054A]
3745 GOSUB 3770:IF flag THEN 3820          [B146]
3746 IF fl THEN 3745                       [A26E]
3747 RETURN                                [87AA]
3770 in$=INKEY$:IF in$="" THEN RETURN      [8C02]
3780 IF in$=" " THEN IF fl THEN fl=0:RET
     URN ELSE fl=1:GOTO 3770               [B304]
3790 IF ASC(in$)=13 THEN flag=1:RETURN E
     LSE flag=0                            [E1CE]
3800 RETURN                                [0196]
3810 PRINT:PRINT:PRINT"{2 SPACE}{CTRL X}
     press enter key to return{CTRL X}":
     RETURN                                [50D8]
3820 MODE 0:CALL prscreen:GOTO 280         [C3CE]
3900 '                                     [C526]
3910 '  ###### c i r c l e ######          [3B94]
3920 '                                     [C32A]
3921 POKE &8B44,1:POKE &8B4D,&19:GOSUB 4
     80:BORDER 4                           [0284]
3925 GOSUB 3930:IF rfl THEN CALL curcl,a
     dro:GOTO 280                          [93A4]
3926 GOTO 3995                             [A14E]
3930 CALL prscreen:CALL curset,adro        [F4D6]
3940 GOSUB 4000:IF rfl THEN RETURN         [81A0]
3942 IF r1fl THEN 3930                     [2FA8]
3950 GOSUB 4100:IF r1fl THEN 3930          [0BE4]
3960 x0=xpoia+xpoir:y0=ypoia-ypoir         [E4EE]
3965 CALL curcl,adro:PLOT x0,y0,color:CA
     LL curset,adro                        [DBBA]
3970 GOSUB 4000:IF rfl THEN RETURN         [C4A6]
3975 IF r1fl THEN 3930                     [E4B4]
3980 GOSUB 4100:IF r1fl THEN 3930          [FBEA]
3990 x1=xpoia+xpoir:y1=ypoia-ypoir:fl=0:
     RETURN                                [2C1E]
3995 r=SQR((x1-x0)*(x1-x0)+(y1-y0)*(y1-y
     0))                                   [CC5E]
3996 CALL curcl,adro:PLOT xpoia+xpoir,yp
     oia-ypoir,col:GOTO 5000               [6752]
4000 in$=INKEY$:IF in$="" THEN 4000        [DBB0]
4010 rfl=0:IF ASC(in$)=13 THEN rfl=1:RET
     URN                                   [1BFC]
4020 r1fl=0:IF ASC(in$)=32 THEN r1fl=1:R
     ETURN                                 [1EC4]
4040 kfl=0:GOSUB 1150:IF kfl THEN GOSUB
     1221                                  [875C]
4050 IF ASC(in$)=88 OR ASC(in$)=224 THEN
      RETURN                               [B220]
4060 GOTO 4000                             [A00E]
4090                                       [E728]
4100 ndfl=1:CALL curcl,adro:GOSUB 1080     [BFF6]
4110 xpoia=xc*16:ypoia=(24-yc)*16+15       [BD56]
4130 GOSUB 920                             [1D46]
4140 IF TEST(xpoia+xpoir,ypoia-ypoir) TH
     EN col=0 ELSE col=color               [0AE4]
4150 PLOT xpoia+xpoir,ypoia-ypoir,col      [3BBE]
4170 in$=INKEY$:IF in$="" THEN 4170        [3FD0]
4180 IF ASC(in$)=3 THEN GOSUB 1260:GOTO
     4170                                  [6FCC]
4190 r1fl=0:IF ASC(in$)=32 THEN r1fl=1:R
     ETURN                                 [D7D4]
4200 keyfl=0:GOSUB 1020:IF keyfl THEN GO
     SUB 880:GOTO 4130                     [C032]
4210 IF ASC(in$)=88 OR ASC(in$)=224 THEN
      RETURN                               [A41C]
4220 GOTO 4170                             [A31A]
4260 '                                     [9A26]
5000 DEG:fl=0:flag=0:chrzei=adrchr+newch
     r*16                                  [2F44]
5005 x=0:xp=x0+r*COS(x):yp=y0-r*SIN(x):o
     x=xp\16:oy=24-yp\16:ox1=ox:oy1=oy     [29CA]
5006 PLOT xp,yp,color                      [F224]
5010 FOR x=0 TO 360                        [A49C]
5020    xp=x0+r*COS(x):yp=y0-r*SIN(x)      [311C]
5025    IF xp<0 OR xp>639 OR yp<0 OR yp>
     399 THEN flag=1:GOTO 5060             [6660]
5030    IF flag THEN PLOT xp,yp,color:fl
     ag=0 ELSE DRAW xp,yp,color            [FCCE]
5040    GOSUB 5800                         [1B6C]
5050    IF INKEY(54)=128 THEN GOSUB 6120
     :GOTO 3925                            [3C92]
5060 NEXT                                  [DF54]
5070 GOSUB 5840:CALL prscreen:CALL curse
     t,adro:PLOT x0,y0,color:GOTO 3925     [FDB4]
5800 tx=xp\16:ty=24-yp\16                  [62C8]
5810 IF ox<>tx OR oy<>ty THEN IF fl THEN
      GOSUB 5840 ELSE CALL curset,bladr+
     ox*2+oy*80:ox1=ox:oy1=oy:fl=1:IF 1f
     l THEN 5840                           [B870]
5820 ox=tx:oy=ty                           [2F96]
5830 RETURN                                [CBA0]
5840 CALL curcl,bladr+ox1*2+oy1*80         [15D8]
5850 adrn=bladr+ox*2+oy*80                 [FFD8]
5860 ox1=ox:oy1=oy                         [2E4E]
5862 POKE offs,0:POKE offs+1,&8:CALL com
     pare,newchr,adrn:IF PEEK(cfl)<>0 TH
     EN POKE picture+ox+40*oy,newchr-PEE
     K(cfl):GOTO 5915                      [01D2]
5865 IF newchr=256 THEN GOSUB 9030:CALL
     prscreen:GOTO 280                     [EFD0]
5870 FOR i=0 TO 7                          [41D6]
5880    POKE chrzei+i,PEEK(adrn+i*2048)    [4E20]
5890    POKE chrzei+i+8,PEEK(adrn+i*2048
     +1)                                   [FDA0]
5900 NEXT                                  [3A5A]
5905 POKE picture+ox+40*oy,newchr          [4548]
5910 newchr=newchr+1:chrzei=chrzei+16      [8B52]
5915 CALL curset,adrn                      [BA2E]
5920 RETURN                                [A6A0]
6000 '                                     [C61A]
6010 '  ###### d r a w   l i n e ######    [6E50]
```

Listing 1. Steuerprogramm für »Background Painter« (Fortsetzung)

```
6020 '                                        [C81E]
6025 POKE &8B44,1:POKE &8B4D,&19:rfl=0:n
     dfl=1:GOSUB 480:BORDER 3                 [D6EC]
6030 GOSUB 3930:IF rfl THEN CALL curcl,a
     dro:GOTO 280                             [C790]
6040 PLOT x0,y0,col:d=0:xp=x0:yp=y0:lfl=
     1                                        [BF62]
6045 ox=xp\16:oy=24-yp\16:ox1=ox:oy1=oy      [AF94]
6050 dx=x1-x0:xr=SGN(dx):dx=ABS(dx)          [F59C]
6060 dy=y1-y0:yr=SGN(dy):dy=ABS(dy)          [77AC]
6065 chrzei=adrchr+newchr*16                  [1FD4]
6070 FOR j=1 TO dx+dy                         [6126]
6080   IF d<=0 THEN d=d+dx:PLOTR 0,yr:yp
     =yp+yr ELSE d=d-dy:PLOTR xr,0:xp=xp
     +xr                                      [CDF6]
6085   GOSUB 5800                             [7040]
6086   IF INKEY(54)=128 THEN GOSUB 6120:
     GOTO 6030                                [F952]
6090 NEXT                                     [E55C]
6100 GOSUB 5840:GOTO 6030                     [E768]
6120 LOCATE 1,1:PEN 3:PRINT"{CTRL X}{7 S
     PACE}break{2 SPACE}!!{4 SPACE}{CTRL
      X}":FOR w=1 TO 2000:NEXT:RETURN         [0D7E]
6200 '                                        [CA1E]
6210 '   ###### f i l l ######                [5176]
6220 '                                        [C822]
6230 BORDER 9                                 [FC04]
6240 rfl=0:GOSUB 2530:IF rfl THEN actchr
     =temp:POKE chrnum,actchr:GOTO 280 E
     LSE POKE clbyte,&3E:POKE clbyte+1,a
     ctchr:POKE xcc,tx:POKE ycc,ty:CALL
     lbuff,picture+tx+40*ty:POKE clbyte,
     &1A:POKE clbyte+1,0:GOTO 6240           [9AA8]
6300 '                                        [CC20]
6310 '   ###### h e l p ######                [1C7C]
6320 '                                        [EE24]
6330 MODE 2:PEN 1                             [3BA2]
6340 PRINT"{13 SPACE}###### i n f o r m
     a t i o n{2 SPACE}######"                [CD90]
6350 PRINT"{12 SPACE}--------------------
     ---------------------"                   [51F0]
6360 PRINT:PRINT                              [8986]
6370 PRINT"character-set :":PRINT"starta
     dress : &";HEX$(adrchr):PRINT"endad
     ress : &";HEX$(adrchr+newchr*16-1):
     LOCATE 25,5:PRINT"number of charact
     ers :";newchr:LOCATE 25,6:PRINT"cha
     rcters free :";256-newchr                [A17E]
6380 LOCATE 53,5:PRINT"picture :"             [7CE6]
6390 LOCATE 53,6:PRINT"startadress : &";
     HEX$(picture):LOCATE 53,7:PRINT"end
     adress : &";HEX$(picture+40*25-1)        [A9A0]
6395 PRINT:PRINT"{CTRL X}Paint/Zoom:{CTR
     L X}":PRINT"{4 SPACE}() only paint"
     :PRINT"{4 SPACE}{} only zoom"            [FD1A]
6400 PRINT" (CTRL T) character table":PR
     INT"{2 SPACE}CTRL G{2 SPACE}get cha
     racter":PRINT" (CTRL N) get code":P
     RINT"{2 SPACE}CTRL Z{2 SPACE}zoom":
     PRINT"{2 SPACE}CTRL D{2 SPACE}defin
     e chr.":PRINT"{2 SPACE}CTRL R{2 SPA
     CE}redefine chr.":PRINT"{2 SPACE}sp
     ace{3 SPACE}clr. chr."                   [3302]
6405 WINDOW #1,29,80,11,25:PAPER #1,0:PE
     N #1,1:CLS #1:WINDOW SWAP 0,1           [B742]
6406 PRINT"Definition:"                       [2C08]
6410 PRINT"{2 SPACE}{CTRL S} search chr.
     ":PRINT"{3 SPACE}CTRL C{2 SPACE}get
      color":PRINT"{3 SPACE}CTRL B{2 SPA
     CE}break def.":PRINT"{3 SPACE}clr{5
      SPACE}clear def.":PRINT"{3 SPACE}E
     nter{3 SPACE}take def.":PRINT"{3 SP
     ACE}CTRL Z{2 SPACE}zoom chr."            [7250]
6420 WINDOW SWAP 0,1:WINDOW #0,54,80,12,
     25                                       [CE38]
6425 PRINT "{CTRL O}{6 SPACE}ovly chr."       [8B26]
6430 PRINT "{CTRL ";CHR$(1);"{CTRL K} ("
     ;CHR$(1);"{CTRL J})}{2 SPACE}mirror
      x":PRINT"{CTRL ";CHR$(1);"{CTRL I}
      (";CHR$(1);"{CTRL H})}{2 SPACE}mir
     ror y":PRINT"{shift ";CHR$(1)"{CTRL
      K}";CHR$(1);"{CTRL J}";CHR$(1);"{C
     TRL H}";CHR$(1);"{CTRL I}}{2 SPACE}
     move chr."                               [8210]
6440 PRINT "{space}{7 SPACE}clr. point"       [2CF8]
6450 WINDOW #0,1,80,20,25                     [6298]
6460 PRINT"{CTRL X}line/circle:{CTRL X}"
                                              [86E8]
6470 PRINT:PRINT"{2 SPACE}space{3 SPACE}
     set marks":PRINT"{2 SPACE}CTRL B{2
     SPACE}break "                            [240C]
6480 LOCATE 32,1:PRINT "{CTRL X}move/cop
     y/fill:{CTRL X}":LOCATE 34,3:PRINT
     "space{2 SPACE}set marks"                [66E2]
6900 PRINT:PRINT:GOSUB 7785:MODE 0:CALL
     prscreen:GOTO 280                        [2258]
7000 '                                        [C31C]
7010 '   ###### g a r b a g e   c o l l e
     c t i o n ######                         [55D0]
```

```
7020 '                                        [C520]
7030 PEN 3:LOCATE 1,1:PRINT"{CTRL X} gar
     bage collection {CTRL X}";:PEN 1 :P
     RINT STRING$(20,"-")                     [CC96]
7040 WINDOW #1,4,17,4,10:PAPER #1,7:CLS
     #1                                       [1B62]
7050 PEN #1,0:PRINT#1:PRINT#1:PRINT#1,"
     are you sure":PRINT#1,"{5 SPACE}y/n
     "                                        [5DAA]
7060 in$=INKEY$:IF in$="y" THEN 7070 ELS
     E IF in$="n" THEN CALL prscreen:GOT
     O 280 ELSE GOTO 7060                     [BF5E]
7070 GOSUB 480:WINDOW #1,1,20,3,INT((new
     chr-1)/20)+4:PAPER #1,0:CLS #1           [0334]
7071 PEN 1:LOCATE 1,INT((newchr-1)/20)+4
     :PRINT STRING$(20,"-")                   [3D4C]
7072 IF newchr>1 THEN CALL prtable,newch
     r ELSE GOTO 7139                         [AEE6]
7075 FOR count=newchr-1 TO 1 STEP -1          [5E66]
7076   adro1=bladr+count*4+160:CALL curs
     et,adro1                                 [DBA0]
7080   CALL abfall,count                      [D344]
7090   IF PEEK(cfl)=0 THEN CALL curcl,ad
     ro1:GOTO 7138                            [C048]
7100   count=PEEK(cfl)                        [7C72]
7110   IF count <> newchr-1 THEN CALL mo
     chr,(newchr-1-count)*16,count*16+ad
     rchr                                     [9564]
7120   CALL chpic,count                       [4784]
7130   newchr=newchr-1:IF newchr>1 THEN
     CALL prtable,newchr                      [8BF6]
7138 NEXT                                     [F264]
7139 GOSUB 490:CALL prscreen:GOTO 280         [416C]
7140 '                                        [9D26]
7150 '   ###### d i s c ######                [CC76]
7160 '                                        [9B2A]
7170 CALL curcl,madr                          [8028]
7180 GOSUB 490:LOCATE 1,1:PEN 2:PRINT"{C
     TRL X}{5 SPACE}disc menu{6 SPACE}{C
     TRL X}";                                 [AA14]
7190 PEN 1:PRINT STRING$(20,"-")              [A0AE]
7200 WINDOW #1,3,17,4,25:PAPER #1,7:WIND
     OW SWAP 0,1                              [367C]
7210 xc=6:yc=3:adro=bladr+xc*2+yc*80          [1DB6]
7220 CLS:PEN 3                                [B27A]
7230 PRINT                                    [CCF2]
7240 PRINT" load file":PRINT:PRINT" save
      all":PRINT:PRINT" save picture":PR
     INT:PRINT" save table"                   [907C]
7250 PRINT:PRINT" load table ap":PRINT:P
     RINT" show picture":PRINT:PRINT" sh
     ow table":PRINT:PRINT" catalog":PRI
     NT:PRINT" menu":PEN 0:PRINT:PRINT"-
     ------------";:PRINT" select item!"
                                              [A6B6]
7260 CALL curset,adro                         [AA26]
7270 '                                        [9A2E]
7280 in$=INKEY$:IF in$="" THEN 7280           [40E4]
7290 kfl=0:GOSUB 1150:IF kfl THEN GOSUB
     1221:GOTO 7280                           [8A38]
7300 IF ASC(in$)=88 OR ASC(in$)=224 THEN
     7330                                     [CFFC]
7310 GOTO 7280                                [832A]
7320 '                                        [BF26]
7330 IF xc<4 OR xc>31 OR yc<3 OR yc>20 T
     HEN 7280                                 [9C1C]
7340 x=(yc-3)/2+0.5                           [C79E]
7350 IF x<>INT(x) THEN 7280                   [A508]
7360 ON x GOTO 7400,7480,7530,7680,7900,
     7800,7810,7450,7380                      [8DA2]
7370 '                                        [9A30]
7380 WINDOW SWAP 0,1:GOSUB 490:CALL prsc
     reen:GOTO 280                            [169C]
7390 '                                        [C434]
7400 CLS:PEN 1:PRINT"{CTRL X}{3 SPACE}lo
     ad file{3 SPACE}{CTRL X}"                [882C]
7410 GOSUB 7750:IF a$="" THEN 8000            [9D0A]
7415 POKE buffer+11,0:POKE adrchr,newchr
     -1                                       [3A62]
7420 LOAD ""+a$                               [4E42]
7421 IF PEEK(buffer+11)=255 THEN 7425         [D768]
7422 newchr=PEEK(adrchr)+1:actchr=1:POKE
      chrnum,actchr:POKE adrchr,0:GOTO 7
     220                                      [C708]
7425 LOCATE 1,10:PRINT"part of picture":
     PEN 3:PRINT"enter:":PEN 0:PRINT:PRI
     NT" {CTRL X}o{CTRL X}riginal or":PR
     INT" {CTRL X}n{CTRL X}ew":PRINT:PRI
     NT"{2 SPACE}position"                    [B47E]
7430 POKE spx,PEEK(buffer+14):POKE spy,P
     EEK(buffer+15)                           [44AC]
7431 in$=INKEY$:IF in$="o" THEN 7434 ELS
     E IF in$="n" THEN 7438 ELSE GOTO 74
     31                                       [8770]
7434 adr=picture+PEEK(buffer+12)+40*PEEK
     (buffer+13):POKE xcc,PEEK(buffer+12
     ):POKE ycc,PEEK(buffer+13)               [7E4E]
7435 GOSUB 480                                [2C5E]
7436 POKE adrchr,0:CALL lbuff,adr:CALL p
     rscreen:GOSUB 7790:WINDOW SWAP 0,1:
```

```
     GOTO 7180                                 [CF02]
7438 POKE adrchr,0:GOSUB 480:CALL prscre
     en:CALL curset,adro                       [0D82]
7440 in$=INKEY$:IF in$="" THEN 7440            [E6DC]
7442 kfl=0:GOSUB 1150:IF kfl THEN GOSUB
     1221:GOTO 7440                            [2732]
7443 IF in$="X" OR ASC(in$)=224 THEN 744
     5                                         [F422]
7444 GOTO 7440                                 [8A36]
7445 POKE xcc,xc:POKE ycc,yc:CALL lbuff,
     picture+xc+40*yc                          [B640]
7446 CALL prscreen:GOSUB 7790:WINDOW SWA
     P 0,1:GOTO 7180                           [2982]
7448 '                                         [B73C]
7450 MODE 2:PEN 1:PAPER 0:CAT                  [73D2]
7460 GOSUB 7785:MODE 0:CALL prscreen:GOT
     O 7180                                    [B8AC]
7470 '                                         [9A32]
7480 CLS:PEN 1:PRINT"{CTRL X}{3 SPACE}sa
     ve all{4 SPACE}{CTRL X}"                  [77CC]
7490 GOSUB 7750:IF a$="" THEN 8000             [551A]
7500 POKE adrchr,newchr-1                      [AB80]
7510 SAVE a$,b,adrcol,16+1000+newchr*16:
     POKE adrchr,0:GOTO 7220                   [47D2]
7520 '                                         [972A]
7530 CLS:PEN 1:PRINT "{CTRL X} save pict
     ure{2 SPACE}{CTRL X}"                     [818A]
7540 PEN 0:PRINT " {CTRL X}a{CTRL X}ll o
     r {CTRL X}p{CTRL X}art"                   [BBBE]
7550 in$=INKEY$:IF in$="" THEN 7550            [AAE4]
7560 IF in$="p" THEN 7590 ELSE IF in$="a
     " THEN 7570 ELSE GOTO 7550                [0C36]
7570 GOSUB 7750:IF a$="" THEN 8000             [8B18]
7580 SAVE a$,b,picture,1000:GOTO 7220          [7D92]
7590 GOSUB 7750:IF a$="" THEN 8000 ELSE
     WINDOW SWAP 0,1                           [37AE]
7600 rfl=0:GOSUB 2530:IF rfl THEN 280 EL
     SE adr=picture+tx+40*ty:CALL copy,a
     dr                                        [5244]
7610 POKE buffer+11,&FF:POKE buffer+12,t
     x:POKE buffer+13,ty:POKE buffer+14,
     PEEK(spx):POKE buffer+15,PEEK(spy)        [1B30]
7615 le=5+PEEK(spx)*PEEK(spy)                  [E35C]
7620 SAVE a$,b,buffer+11,le:CALL prscree
     n:GOTO 7180                               [5378]
7670 '                                         [BA36]
7680 CLS:PEN 1:PRINT"{CTRL X}{2 SPACE}sa
     ve table{3 SPACE}{CTRL X}"                [70EE]
7690 PEN 0:PRINT "{2 SPACE}{CTRL X}a{CTR
     L X}ll or {CTRL X}p{CTRL X}art"           [BC0A]
7700 in$=INKEY$:IF in$="" THEN 7700            [B5D8]
7710 IF in$="p" THEN 7732 ELSE IF in$="a
     " THEN 7720 ELSE GOTO 7700                [8820]
7720 GOSUB 7750:IF a$="" THEN 8000             [B012]
7730 POKE adrchr,newchr-1                      [C18A]
7731 SAVE a$,b,adrchr,newchr*16:POKE adr
     chr,0:GOTO 7220                           [CEDE]
7732 GOSUB 7750:IF a$="" THEN 8000 ELSE
     WINDOW SWAP 0,1                           [4BAA]
7733 GOSUB 480:CALL prscreen:PAPER #1,0:
     GOSUB 1420:GOSUB 1510:IF count=0 TH
     EN GOSUB 490:CALL prscreen::GOTO 71
     80                                        [1CD8]
7734 temp=count:adro=adro1:GOSUB 2700:GO
     SUB 1510:IF count=0 THEN GOSUB 490:
     CALL prscreen:GOTO 7180                   [885E]
7735 IF temp>count THEN GOSUB 9040:GOTO
     7733                                      [AA0A]
7737 adr1=adrchr+temp*16:adr2=adrchr+cou
     nt*16                                     [D278]
7740 temp=PEEK(adr1-1):POKE adr1-1,(adr2
     -adr1)/16+1                               [38BA]
7742 SAVE a$,b,adr1-1,adr2-adr1+17             [4E0A]
7743 POKE adr1-1,temp:GOSUB 490:CALL prs
     creen:GOTO 7180                           [1C3C]
7750 CALL &BB03:PEN 3:PRINT"enter name:"
     :PEN 0:PRINT:INPUT a$                     [94E6]
7770 RETURN                                    [AEAA]
7780 CALL curcl,adro:FOR w=1 TO 100:NEXT
     :CALL curset,adro:RETURN                  [A150]
7785 PRINT"{4 SPACE}press any key to ret
     urn "                                     [295A]
7790 in$=INKEY$:IF in$="" THEN 7790 ELSE
      RETURN                                   [FA8E]
7800 GOSUB 480:CALL prscreen:GOSUB 7790:
     GOSUB 490:WINDOW SWAP 0,1:GOTO 7180
                                               [DE50]
7810 GOSUB 480:WINDOW SWAP 0,1:CLS:LOCAT
     E 1,2:GOSUB 1420:GOSUB 7790:GOSUB 4
     90:CALL prscreen:GOTO 7180                [F90E]
7850 PEN 3:PRINT:PRINT:PRINT"{3 SPACE}se
     t marks{2 SPACE}":FOR w=1 TO 2000:N
     EXT:RETURN                                [9F96]
7900 CLS:PEN 1:PRINT"{CTRL X}{2 SPACE}lo
     ad file ap {CTRL X}"                      [2658]
7910 GOSUB 7750:IF a$="" THEN 8000             [C114]
7920 adr1=adrchr+newchr*16:temp=PEEK(adr
     1-1)                                      [A16E]
7930 LOAD ""+a$,adr1-1:newchr=newchr+PEE
     K(adr1-1):POKE adr1-1,temp                [616C]
7935 IF newchr>255 THEN newchr=255             [22AE]
8000 WINDOW SWAP 0,1:GOTO 7180                 [5756]
9000 '                                         [C920]
9010 '  ****** error meldungen ******          [FD24]
9020 '                                         [C724]
9030 PEN 3:LOCATE 1,1:PRINT "{CTRL X} ch
     aracter set full {CTRL X}":FOR w=1
     TO 1500:NEXT:RETURN                       [2056]
9040 PEN 3:LOCATE 1,1:PRINT "{CTRL X}{4
     SPACE}illegal range{3 SPACE}{CTRL X
     }":FOR w=1 TO 1500:NEXT:RETURN            [99CE]
9050 PEN 3:LOCATE 1,1:PRINT "{CTRL X}{2
     SPACE}already defined{3 SPACE}{CTRL
      X}":FOR w=1 TO 1500:NEXT:RETURN          [81E4]
9100 '                                         [C722]
9110 '  ****** meldungen ******                [3D92]
9120 '                                         [C926]
9130 PEN 1:LOCATE 1,1:PRINT "{CTRL X} ch
     ar. found:{CTRL X}";:PEN 3:PRINT ne
     wchr-PEEK(cfl):FOR w=1 TO 1500:NEXT
     :RETURN                                   [2AA2]
9140 PEN 3:LOCATE 1,1:PRINT "{CTRL X} ch
     ar. not found !! {CTRL X}":FOR w=1
     TO 1500:NEXT:RETURN                       [6838]
10000 '  ****** anfangsbild ******             [E152]
10005 CALL prscreen:GOSUB 480                  [AE54]
10010 LOCATE 5,21:PEN 1:PRINT"press";:PE
      N 2:PRINT" enter":GOTO 11000             [D298]
11000 '  ****** farbwechsel fuer anfangs
      bild ******                              [0B44]
11010 anf=2                                    [C46E]
11020 anf=anf-1:IF anf<2 THEN anf=11           [CBD0]
11030 an=anf                                   [62AC]
11040 FOR i=2 TO 11                            [7374]
11050      INK i,PEEK(adrcol+an):an=an+1:I
      F an>11 THEN an=2                        [C7D2]
11060 NEXT                                     [FFAE]
11070 IF INKEY(18)=0 THEN RETURN ELSE GO
      TO 11020                                 [68E8]
```

Listing 1. Steuerprogramm für »Background Painter« (Schluß)

```
90 ' ########### Assembler-Unterroutinen
   ###########                                 [B432]
95 '                                           [856A]
100  DATA 221,110,0,221,102,1,221,94,2,2
     21,86,3,6,16,126,245,1675                 [0C12]
101  DATA 230,170,32,3,26,230,170,79,241
     ,230,85,32,3,26,230,85,1872               [3A06]
102  DATA 177,18,35,19,16,232,201,205,10
     3,142,6,8,221,126,8,245,1762              [B872]
103  DATA 245,221,126,0,230,170,203,63,7
     9,221,126,0,230,85,203,39,2241            [7E18]
104  DATA 177,221,119,8,241,230,170,203,
     63,79,241,230,85,203,39,177,2486          [FB2A]
105  DATA 221,119,0,221,35,16,213,201,20
     5,103,142,6,8,221,126,0,1837              [B23C]
106  DATA 245,230,170,203,63,221,119,0,2
     41,230,85,203,39,79,221,126,2475          [DE02]
107  DATA 8,230,170,203,63,177,221,119,8
     ,221,35,16,224,201,205,103,2204           [5D7C]
108  DATA 142,6,8,221,126,8,245,230,85,2
     03,39,221,119,8,241,230,2132              [2270]
109  DATA 170,203,63,79,221,126,0,230,85
     ,203,39,177,221,119,0,221,2157            [783C]
110  DATA 35,16,224,201,243,221,110,0,22
     1,102.1,221,33,0,192,6,1826               [42BA]
111  DATA 25,197,221,229,6,5,197,221,229
     ,94,22,0,213,253,225,17,2154              [8186]
112  DATA 160,136,205,168,140,221,225,22
     1,35,221,35,35,193,16,231,17,2259         [095C]
113  DATA 80,0,221,225,221,25,193,16,216
     .251,201,18,17,17,17,20,1738              [F45E]
114  DATA 15,7,6,0,15,19,17,17,17,16,15,
     22,23,21,15,19,244                        [F87C]
115  DATA 17,17,17,16,15,29,28,0,15,19,1
     7,17,17,16,15,8,263                       [EB06]
116  DATA 9,0,15,19,17,17,17,16,15,32,30
     ,31,15,19,17,17,286                       [D4F8]
117  DATA 17,16,15,12,13,0,15,19,17,17,1
     7,16,15,14,13,0,216                       [58CC]
118  DATA 15,19,17,17,17,16.15,10,11,0.1
     5,19,17,17,17,16,238                      [4D50]
119  DATA 15,0,36,0,15,19,17,17,17,16,15
     .2,3,4,15,19,210                          [BCAA]
120  DATA 17,17,17,16,15.2,3,2,15,19,17,
     17,17,16,15,0,205                         [010A]
121  DATA 37,0,15,19,17,17,17,16,15,5,1,
     3,15,19,17,17,230                         [9916]
122  DATA 17,16,15,24,25,0,15,35,26,27,3
     4,33,0,0,0,0,267                          [2098]
123  DATA 0,0,0,0,0,0,0,0,0,0,0,0,64,128
     ,128,192,512                              [28C8]
124  DATA 128,128,128,0,128,64,64,192,64
     .64.64.0.64,128,128,128,1472              [32A8]
```

Listing 2. Basic-Hexlader der Maschinencode-Routinen

```
125  DATA 128,128,64,0,128,0,0,0,0,0,128
     ,0,128,128,128,128,1088              [A96A]
126  DATA 128,128,192,0,0,0,0,0,0,128,
     0,64,128,128,64,960                  [89D2]
127  DATA 0,0,192,0,192,0,0,128,64,64,12
     8,0,128,128,128,128,1280             [10DA]
128  DATA 128,64,64,0,64,64,64,64,64,128
     ,128,0,0,0,136,204,1172              [DD8E]
129  DATA 204,68,0,0,0,0,0,136,204,204
     ,0,0,192,64,0,1072                   [D840]
130  DATA 0,0,0,0,128,192,192,192,0;0,0,
     0,0,4,8,8,724                        [1F5C]
131  DATA 8,4,0,0,12,0,0,0,0,12,0,8,4,
     0,0,48                               [9472]
132  DATA 0,4,8,0,0,0,8,8,8,0,0,0,68,2
     04,204,512                           [D52C]
133  DATA 68,0,0,0,68,204,152,48,204,204
     ,0,0,204,204,48,48,1452              [0266]
134  DATA 48,204,204,0,0,136,204,100,204
     ,136,0,0,4,8,8,8,1264                [8F88]
135  DATA 8,8,8,4,76,140,140,140,140,140
     ,140,76,136,68,68,68,1360            [466A]
136  DATA 68,68,68,136,0,0,0,0,0,0,0,4
     ,0,8,0,352                           [D83C]
137  DATA 8,0,8,4,76,140,136,140,136,140
     ,136,76,16,16,16,16,1064             [16EE]
138  DATA 16,16,16,16,0,0,0,0,0,0,0,16
     ,16,16,16,128                        [B852]
139  DATA 48,16,16,16,0,0,0,0,0,0,0,0,
     0,0,0,96                             [2F4E]
140  DATA 48,0,0,0,0,0,0,48,0,0,0,0,0,
     0,0,96                               [EF6C]
141  DATA 16,16,16,16,0,0,0,48,0,0,0,1
     6,16,16,16,176                       [CDC4]
142  DATA 16,16,16,16,0,0,0,48,0,0,0,0
     ,0,0,0,112                           [1CFA]
143  DATA 48,16,16,16,0,0,0,0,0,0,0,12
     ,12,12,12,144                        [7F30]
144  DATA 12,12,12,12,12,12,12,12,12,12,
     12,12,204,204,204,204,960            [73DC]
145  DATA 204,204,204,204,204,204,204,20
     4,204,204,204,204,192,192,192,192,32
     16                                   [E230]
146  DATA 192,192,192,192,192,192,192,19
     2,192,192,192,192,68,68,68,68,2576   [9962]
147  DATA 68,68,68,68,204,204,204,204,20
     0,200,200,204,204,204,204,196,2700   [25BA]
148  DATA 64,64,64,196,204,204,136,136,2
     04,204,204,204,68,68,68,68,2156      [4BE6]
149  DATA 68,0,0,48,204,204,204,204,204,
     0,0,48,204,68,68,68,1592             [3FFA]
150  DATA 204,0,0,48,204,204,204,204,204
     ,0,0,48,0,0,4,12,1336                [9160]
151  DATA 8,0,0,0,4,12,8,0,0,0,0,0,0,0
     ,0,32                                [B5FA]
152  DATA 0,0,0,0,0,0,0,4,12,8,0,0,64,
     192,208,488                          [7B8A]
153  DATA 208,192,64,0,192,192,240,240,2
     40,240,192,192,0,128,192,224,2736    [3278]
154  DATA 224,192,128,0,0,0,0,0,0,0,0,
     0,0,0,0,544                          [1360]
155  DATA 0,68,204,204,0,0,0,68,204,204,
     136,0,16,16,16,16,1152               [46F0]
156  DATA 16,16,16,48,0,0,0,0,0,0,0,0,
     0,0,0,96                             [C74C]
157  DATA 0,0,0,48,0,0,0,0,0,0,48,16,1
     6,16,16,160                          [F684]
158  DATA 16,16,16,16,0,0,0,0,0,0,48,8
     0,240,176,176,784                    [E81A]
159  DATA 176,176,176,240,160,240,176,17
     6,176,176,240,240,20,40,0,20,2432    [CA7A]
160  DATA 20,20,0,20,40,20,20,40,0,0,0,0
     ,243,62,0,50,535                     [7B4C]
161  DATA 98,139,221,126,0,33,147,146,1,
     232,3,237,161,40,6,234,1824          [7218]
162  DATA 15,139,50,98,139,251,201,221,1
     10,0,221,102,1,221,70,2,1841         [BB40]
163  DATA 229,221,225,62,0,50,98,139,33,
     123,150,197,221,229,229,253,2459     [F534]
164  DATA 225,17,0,8,6,8,221,126,0,253,1
     90,0,32,22,221,126,1455              [2068]
165  DATA 1,253,190,8,32,14,253,35,221,2
     5,16,234,221,225,193,120,2041        [90C0]
166  DATA 50,98,139,201,221,225,193,17,1
     6,0,25,16,206,201,0,221,1829         [3C72]
167  DATA 110,0,221,102,1,17,155,142,58,
     114,142,71,197,229,58,113,1730       [F534]
168  DATA 142,79,6,0,237,176,225,1,40,0,
     9,193,16,238,201,205,1768            [EB70]
169  DATA 103,142,253,33,155,142,17,155,
     142,58,117,142,103,58,114,142,1876   [ECE4]
170  DATA 71,197,221,229,58,116,142,111,
     58,113,142,71,26,0,221,119,1895      [8FB8]
171  DATA 0,125,254,39,40,22,221,35,19,4
     4,16,240,221,225,1,40,1542           [448A]
172  DATA 0,221,9,193,124,254,24,200,36,
     16,214,201,58,113,142,79,1884        [5CE2]
173  DATA 6,0,253,9,253,229,209,24,227,2
     21,70,0,253,33,0,0,1787              [E588]
174  DATA 205,230,139,5,205,222,139,16,2
     51,201,120,50,121,142,253,42,2341    [943A]
175  DATA 121,142,104,38,0,41,41,221,33,
     160,192,235,221,25,197,205,1976      [859E]
176  DATA 165,140,193,201,221,94,0,221,8
     6,1,221,78,2,221,70,3,1917           [2AAC]
177  DATA 33,16,0,25,237,176,201,221,126
     ,0,33,147,146,6,25,197,1589          [C834]
178  DATA 6,40,190,218,34,140,194,35,140
     ,54,0,195,35,140,53,35,1509          [0B28]
179  DATA 16,240,193,16,234,201,221,110,
     0,221,102,1,229,253,225,221,2483     [F0DC]
180  DATA 110,2,221,102,3,229,221,225,19
     5,178,140,221,110,0,221,102,2280     [AEB8]
181  DATA 1,253,33,160,194,195,113,141,2
     05,103,142,253,33,139,142,6,2113     [AAEE]
182  DATA 8,17,0,8,253,126,0,221,119,0,2
     53,126,8,221,119,1,1480              [AD72]
183  DATA 253,35,221,25,16,238,201,205,1
     03,142,17,0,8,253,33,123,1873        [E2CC]
184  DATA 142,6,8,221,126,0,253,119,16,2
     30,85,253,182,0,221,119,1981         [1780]
185  DATA 0,221,126,1,253,119,24,230,170
     ,253,182,8,221,119,1,253,2181        [A3C6]
186  DATA 35,221,25,16,222,201,205,103,1
     42,253,42,121,142,205,165,140,2238   [4DA2]
187  DATA 201,17,123,150,253,41,253,41,2
     53,41,253,41,253,25,17,0,1962        [EAD0]
188  DATA 8,6,8,253,126,0,221,119,0,253,
     126,8,221,119,1,221,1690             [80EA]
189  DATA 25,253,35,16,238,201,33,147,14
     6,6,25,197,6,40,54,0,1422            [2F66]
190  DATA 35,16,251,193,16,245,201,243,2
     21,33,0,192,33,147,146,6,1978        [FCF2]
191  DATA 25,197,6,40,197,221,229,78,6,0
     ,197,253,225,205,165,140,2184        [CB14]
192  DATA 221,225,221,35,221,35,35,193,1
     6,234,193,16,228,251,201,243,2568    [2E78]
193  DATA 205,103,142,253,33,0,192,6,3,1
     97,221,229,253,229,58,118,2242       [F752]
194  DATA 142,50,120,142,205,50,141,253,
     225,17,128,2,253,25,221,225,2199     [5DF8]
195  DATA 17,40,0,221,25,33,119,142,52,1
     93,16,221,251,201,6,5,1542           [0C96]
196  DATA 197,253,229,58,120,142,254,0,2
     50,107,141,254,40,242,107,141,2535   [9ADA]
197  DATA 58,119,142,254,0,250,107,141,2
     54,25,242,107,141,221,126,0,2187     [7704]
198  DATA 205,91,142,205,113,141,253,225
     ,17,16,0,253,25,221,35,33,1975       [6B44]
199  DATA 120,142,52,193,16,202,201,33,1
     23,150,195,87,141,6,2,197,1860       [B54E]
200  DATA 253,229,229,253,229,6,8,253,22
     9,126,230,170,79,203,63,177,2737     [6144]
201  DATA 197,205,192,141,193,35,253,225
     ,17,80,0,253,25,16,232,253,2317      [95AA]
202  DATA 225,17,4,0,253,25,225,6,8,253,
     229,126,230,85,79,203,1968           [84D4]
203  DATA 39,177,197,205,192,141,193,35,
     253,225,17,80,0,253,25,16,2048       [7C64]
204  DATA 232,253,225,17,8,0,253,25,193,
     16,180,201,17,0,8,6,1634             [62DE]
205  DATA 8,253,119,0,253,119,1,253,119,
     2,253,119,3,253,25,16,1796           [3DC4]
206  DATA 240,201,253,33,123,142,253,126
     ,0,253,119,16,253,126,7,253,2398     [D602]
207  DATA 119,23,253,126,8,253,119,24,25
     3,126,15,253,119,31,205,103,2030     [DEF6]
208  DATA 142,6,6,221,126,1,230,85,253,1
     82,1,253,119,17,221,126,1989         [5688]
209  DATA 9,230,170,253,182,9,253,119,25
     ,221,35,253,35,16,228,33,2071        [78EE]
210  DATA 139,142,253,33,160,194,195,113
     ,141,58,121,142,205,91,142,253,2382  [E932]
211  DATA 33,160,194,195,113,141,221,110
     ,0,221,102,1,229,253,225,58,2256     [0CDE]
212  DATA 115,142,79,126,254,0,40,2,14,0
     ,121,205,44,188,195,192,1717         [0176]
213  DATA 141,58,115,142,205,47,188,50,1
     15,142,201,58,115,142,205,44,1968    [C778]
214  DATA 188,230,85,50,115,142,201,111,
     38,0,17,123,150,41,41,41,1573        [91B6]
215  DATA 41,25,201,221,110,0,221,102,1,
     229,221,225,201,0,0,0,1798           [4350]
216  DATA 0,0,0,0,0,0,0,193,8,136,32,160
     ,40,168,143,143,1023                 [600E]
217  DATA 4,68,16,80,20,84,193,0,0,0,0,0
     ,0,0,0,0,465                         [36FE]
218  DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,
     0,0                                  [F30A]
220  DATA 0,24,20,6,26,1,2,8,10,12,14,16
     ,18,22,15,9,203                      [7620]
500 MEMORY 34611:adr=34612                [02B6]
510 FOR l=100 TO 218:sum=0                [1BE2]
520 FOR i=0 TO 15:READ a:POKE adr,a:adr=
    adr+1:sum=sum+a:NEXT                  [E7D0]
530 READ a:IF a<> sum THEN PRINT"error i
    n zeile:"l                            [D69C]
540 NEXT                                  [75F0]
550 FOR i=0 TO 15:READ a:POKE 37507+i,a:
    NEXT                                  [D3C0]
560 FOR i=38523 TO 38523+16:POKE i,0:NEX
    T                                     [A45C]
570 SAVE "bp.bin",b,34612,3927            [9186]
```

Listing 2. Basic-Hexlader der Maschinencode-Routinen (Schluß)

```
100 '###### anfangsgrafik ######                 [AA1C]
110 '                                            [DDB2]
120 MEMORY 34611:LOAD"bp.bin",34612              [885C]
550 adr=37507                                    [1868]
560 FOR l=600 TO 872                             [A60A]
565   sum=0                                      [E584]
570   FOR i=0 TO 14                              [6B42]
575   READ a:POKE adr,a:adr=adr+1:sum=su
    m+a                                          [35B2]
580   NEXT:READ a:IF a<>sum THEN PRINT"e
    rror in Zeile:"l:END                         [CFBA]
590 NEXT                                         [71FA]
595 SAVE"bp.bin",b,34612,6983:END                [3880]
600 DATA 0,26,9,12,18,19,10,13,6,12,14,1
    7,16,22,2,196                                [0EA6]
601 DATA 3,0,0,0,0,0,0,0,0,0,0,0,0,0,3
                                                 [AB16]
602 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [1E0C]
603 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [9B0E]
604 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [AC10]
605 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [A912]
606 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [5214]
607 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [1F16]
608 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [9018]
609 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [5D1A]
610 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [9C0A]
611 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,122
    ,122                                         [6AA0]
612 DATA 123,123,123,123,123,123,123,123
    ,123,123,123,123,123,123,123,1845            [EB46]
613 DATA 123,123,123,123,123,123,123,123
    ,123,123,123,123,123,123,123,1845            [0D48]
614 DATA 122,0,0,0,0,0,0,0,124,0,0,0,0
    ,0,246                                       [8382]
615 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [5314]
616 DATA 0,0,0,0,0,0,0,0,0,0,125,0,0,0,0
    ,125                                         [C1B6]
617 DATA 0,0,0,0,124,0,1,4,0,8,9,15,16,1
    8,23,218                                     [7F9E]
618 DATA 29,30,32,35,36,23,4,0,46,45,47,
    48,54,56,59,544                              [D5C6]
619 DATA 64,65,71,69,0,125,0,0,0,0,0,0,0
    ,0,124,518                                   [CA6E]
620 DATA 0,3,6,22,21,10,14,20,0,24,28,27
    ,33,40,39,287                                [7B9C]
621 DATA 24,28,27,43,44,50,49,55,57,60,6
    3,66,73,70,0,709                             [6226]
622 DATA 125,0,0,0,0,0,0,0,124,0,2,5,7
    ,11,274                                      [B708]
623 DATA 12,13,17,19,31,25,26,34,37,38,3
    1,25,26,41,42,417                            [FC56]
624 DATA 51,52,53,58,61,62,67,68,72,0,12
    5,0,0,0,0,669                                [1CD8]
625 DATA 0,0,0,0,124,0,0,0,0,0,0,0,0,0,0
    ,124                                         [70B2]
626 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [AE18]
627 DATA 0,0,0,0,0,125,0,0,0,0,0,0,0,0,1
    24,249                                       [1296]
628 DATA 0,0,0,0,0,0,139,140,0,142,143,1
    44,143,145,0,996                             [DAC4]
629 DATA 146,147,147,145,148,149,150,0,0
    ,0,0,0,0,0,0,1032                            [6A26]
630 DATA 125,0,0,0,0,0,0,0,124,0,0,0,0
    ,0,249                                       [698A]
631 DATA 0,0,141,0,0,0,0,0,0,0,0,0,0,0,0
    ,141                                         [1AA8]
632 DATA 0,0,0,0,157,172,166,167,0,0,125
    ,0,0,0,0,787                                 [A132]
633 DATA 0,0,0,0,124,0,0,0,0,0,109,108
    ,107,0,448                                   [1438]
634 DATA 0,94,0,0,0,0,85,84,0,0,0,0,0,15
    8,159,580                                    [A716]
635 DATA 156,171,163,164,0,125,0,0,0,0,0
    ,0,0,0,124,903                               [ABE2]
636 DATA 0,0,0,0,0,0,110,0,106,111,97,10
    2,91,93,74,784                               [8C00]
637 DATA 75,76,83,81,80,0,0,0,160,162,16
    9,168,170,165,0,1389                         [85B6]
638 DATA 125,0,0,0,0,0,0,0,124,0,0,0,0
    ,0,249                                       [F19A]
639 DATA 0,112,105,0,100,98,95,96,92,0,0
    ,77,82,78,79,1014                            [B482]
640 DATA 87,0,0,0,161,0,154,155,0,0,125,
    0,0,0,0,682                                  [F9B8]
641 DATA 0,0,0,0,124,0,0,0,0,0,0,0,104
    ,103,331                                     [5140]
642 DATA 99,101,0,116,117,0,0,0,90,86,88
    ,89,0,0,0,786                                [80D8]
```

Listing 3. Der Basic-Lader erzeugt das Titelbild

```
643 DATA 0,153,152,0,0,125,0,0,0,0,0,0,0
    ,0,124,554                                   [FF32]
644 DATA 0,0,0,0,0,0,0,113,0,0,0,0,114
    ,115,342                                     [044E]
645 DATA 0,0,0,178,179,180,183,185,186,1
    89,0,0,151,0,0,1431                          [B824]
646 DATA 125,0,0,0,0,0,0,0,124,0,0,0,0
    ,0,249                                       [FD98]
647 DATA 0,0,0,0,0,0,0,118,120,0,0,0,1
    91,181,610                                   [4130]
648 DATA 182,184,188,187,190,0,0,173,136
    ,0,125,0,0,0,0,1365                          [051E]
649 DATA 0,0,0,0,124,0,133,135,138,0,0,0
    ,0,0,0,530                                   [A638]
650 DATA 0,0,131,119,121,130,0,0,176,0,0
    ,0,0,0,0,677                                 [5606]
651 DATA 134,135,175,174,0,125,0,0,0,0,0
    ,0,0,0,124,867                               [7BF0]
652 DATA 0,133,133,133,137,127,127,127,1
    27,127,127,127,128,133,133,1819              [83DA]
653 DATA 129,127,127,177,127,127,127,127
    ,127,132,133,133,133,133,0,1859              [7DEC]
654 DATA 125,0,0,0,0,0,0,0,124,0,0,0,0
    ,0,249                                       [4196]
655 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [231C]
656 DATA 0,0,0,0,0,0,0,0,0,0,125,0,0,0,0
    ,125                                         [31BE]
657 DATA 0,0,0,0,122,126,126,126,126,126
    ,126,126,126,126,126,1382                    [D15A]
658 DATA 126,126,126,126,126,126,126,126
    ,126,126,126,126,126,126,126,1890            [65B4]
659 DATA 126,126,126,126,126,122,0,0,0,0
    ,0,0,0,0,0,752                               [7EE4]
660 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [5814]
661 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [2516]
662 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [B218]
663 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [AF1A]
664 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [241C]
665 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [A11E]
666 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [BE20]
667 DATA 0,0,0,0,0,0,0,0,0,0,0,191,0,0,0
    ,191                                         [10CE]
668 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,64,64
    ,128                                         [27E2]
669 DATA 132,132,132,132,132,192,152,152
    ,152,152,152,152,152,132,132,2180            [B878]
670 DATA 132,132,132,64,64,0,152,152,152
    ,152,152,152,152,192,132,1912                [72F0]
671 DATA 132,132,132,132,132,132,132,152
    ,152,152,152,152,152,152,152,2140            [FF5A]
672 DATA 128,224,180,180,180,180,180,180
    ,0,0,128,128,232,232,232,2384                [9004]
673 DATA 232,180,180,180,180,180,180,224
    ,128,232,232,232,232,128,128,2848            [27AC]
674 DATA 0,0,180,180,180,192,192,180,180
    ,180,232,128,128,0,0,1952                    [5B76]
675 DATA 128,128,232,0,0,0,0,0,0,0,64,
    129,129,129,939                              [2D80]
676 DATA 129,129,129,64,0,0,64,64,133,13
    3,133,133,0,192,142,1445                     [8120]
677 DATA 142,142,142,142,142,0,128,200,2
    00,152,152,152,152,0,0,1846                  [B20E]
678 DATA 0,0,128,128,128,128,152,152,152
    ,152,152,152,152,152,224,1952                [7310]
679 DATA 224,224,224,224,224,224,224,135
    ,135,194,194,194,194,194,128,2936            [DFE6]
680 DATA 192,128,0,0,0,0,0,144,144,64,
    64,64,64,64,928                              [1594]
681 DATA 0,180,180,180,180,180,180,180,1
    92,128,128,128,128,128,128,2220              [5DE0]
682 DATA 128,0,64,64,64,64,64,0,0,0,0,0,
    0,0,0,448                                    [C316]
683 DATA 0,0,0,64,64,64,64,64,64,64,64,0
    ,0,0,0,512                                   [B38E]
684 DATA 0,0,0,0,0,0,0,64,64,64,64,64,64
    ,129,129,642                                 [7160]
685 DATA 169,169,169,169,169,192,135,135
    ,135,135,135,194,128,169,169,2372            [7622]
686 DATA 169,169,169,129,129,64,128,194,
    135,135,135,135,135,192,0,2018               [5DC2]
687 DATA 128,202,202,142,142,142,192,0,0
    ,0,0,128,128,128,128,1662                    [A264]
688 DATA 192,142,142,142,202,202,128,0,1
    28,128,128,128,0,0,0,1662                    [E666]
689 DATA 0,169,169,169,169,169,169,169,1
    69,128,128,128,128,128,128,2120              [B458]
690 DATA 128,128,135,135,135,135,135,135
    ,135,135,142,142,142,142,142,2046            [4194]
691 DATA 142,142,142,0,0,0,0,0,0,0,64,
    64,64,64,682                                 [AA36]
692 DATA 64,64,64,64,0,0,0,64,64,64,64,6
    4,64,144,144,928                             [8130]
693 DATA 152,152,152,152,152,64,64,64,64
```

```
    ,64,64,64,64,152,152,1576                [D6B6]
694 DATA 152,152,152,152,152,152,224,224
    ,224,224,224,192,128,0,169,2521          [66D4]
695 DATA 129,129,64,64,64,0,0,135,135,13
    5,135,135,135,133,64,1457                [68A0]
696 DATA 128,128,128,128,128,128,128,0,0
    ,0,0,0,128,194,194,1412                  [FBBE]
697 DATA 135,0,0,0,0,0,0,128,180,224,2
    24,180,180,180,1431                      [72EA]
698 DATA 180,180,232,128,0,192,169,169,1
    69,169,0,128,192,224,224,2356            [BE58]
699 DATA 180,180,180,0,0,64,129,169,169,
    169,169,0,192,194,135,1930               [7D2E]
700 DATA 135,135,135,192,0,0,128,128,128
    ,128,128,0,64,64,64,1429                 [9222]
701 DATA 64,64,64,0,0,152,152,152,152,15
    2,144,144,64,0,0,1304                    [3AC0]
702 DATA 0,64,64,64,64,64,64,132,132,142
    ,142,142,142,142,64,1422                 [000A]
703 DATA 64,64,64,64,64,64,64,142,142,14
    2,142,142,142,142,142,1584               [A0F0]
704 DATA 64,64,64,64,64,0,0,0,142,142,14
    2,142,142,132,132,1294                   [F834]
705 DATA 64,192,152,152,152,152,152,152,
    152,192,180,180,180,180,180,2412         [2C34]
706 DATA 180,192,0,128,232,169,169,169,1
    69,129,0,0,0,128,128,1793                [2FB0]
707 DATA 128,128,128,144,144,200,152,152
    ,152,152,192,180,224,192,180,2448        [2AAA]
708 DATA 180,180,180,192,169,169,169,169
    ,169,169,232,128,128,128,128,2490        [690C]
709 DATA 128,128,128,0,0,64,0,0,0,128,23
    2,169,169,128,0,1274                     [CA88]
710 DATA 0,0,0,0,128,128,200,200,128,128
    ,128,192,144,144,0,1520                  [886E]
711 DATA 0,0,0,192,180,180,180,64,64,64,
    64,0,0,0,0,988                           [341E]
712 DATA 142,142,142,142,132,132,64,0,15
    2,152,152,152,152,152,192,2000           [FF30]
713 DATA 128,224,224,224,224,128,128,0,0
    ,64,64,64,64,64,64,1664                  [03CE]
714 DATA 64,64,142,142,142,142,142,142,1
    42,142,152,152,152,152,152,2024          [EBAC]
715 DATA 152,152,152,224,224,224,224,224
    ,224,224,224,128,192,152,152,2872        [709A]
716 DATA 152,152,152,152,0,0,128,128,224
    ,224,224,224,0,0,0,1760                  [CA80]
717 DATA 0,64,64,64,64,0,64,132,132,142,
    142,142,142,0,0,1152                     [A260]
718 DATA 0,0,0,0,0,0,0,64,64,148,148,148
    ,148,148,0,868                           [DE32]
719 DATA 128,128,232,232,232,232,232,0,0
    ,0,64,64,64,64,64,1736                   [5B56]
720 DATA 232,232,232,232,232,232,232,232
    ,64,64,64,64,64,64,64,2304               [BCDC]
721 DATA 64,0,0,0,0,0,0,0,148,148,148,
    148,148,148,952                          [4A80]
722 DATA 148,148,0,0,0,0,0,0,0,148,148
    ,148,148,64,952                          [9282]
723 DATA 64,0,0,232,169,169,169,169,169,
    129,64,192,133,135,135,1929              [B196]
724 DATA 135,135,135,192,142,142,142,142
    ,202,202,128,0,128,128,128,2081          [A2BA]
725 DATA 128,0,0,0,0,0,192,192,142,142,1
    42,142,142,0,0,1222                      [9FD2]
726 DATA 0,128,128,128,128,128,142,142,1
    42,142,142,142,142,142,128,1904          [DECA]
727 DATA 128,128,128,128,128,128,128,0,6
    4,64,64,64,64,64,64,1344                 [735A]
728 DATA 192,152,152,152,152,152,152,152
    ,64,64,64,64,64,64,64,1704               [E40A]
729 DATA 64,152,152,152,152,152,152,152,
    152,64,64,64,64,64,64,1664               [690E]
730 DATA 64,0,152,152,152,152,152,152,15
    2,192,0,128,128,128,128,1832             [48A0]
731 DATA 224,224,224,0,64,64,64,64,64,64
    ,64,180,180,180,180,1840                 [FB30]
732 DATA 180,180,180,148,192,192,232,169
    ,169,169,169,169,148,192,192,2681        [CE0E]
733 DATA 128,128,128,128,0,169,169,169,1
    29,129,64,64,0,135,135,1675              [4590]
734 DATA 135,135,135,135,135,192,128,128
    ,128,128,128,128,128,0,135,1898          [7F0A]
735 DATA 135,135,135,135,135,135,135,128
    ,128,128,128,128,128,128,128,1969        [F5DA]
736 DATA 192,135,135,135,135,135,135,135
    ,0,128,128,128,128,128,128,1905          [1AF4]
737 DATA 128,0,0,0,64,64,64,64,64,64,132
    ,132,142,142,142,1202                    [1AD8]
738 DATA 142,142,64,64,64,64,64,64,64,64
    ,142,142,142,142,142,1506                [079A]
739 DATA 142,142,142,64,64,64,64,64,0,0,
    0,142,142,142,142,1314                   [243A]
740 DATA 142,132,132,64,152,152,152,152,
    152,152,152,192,180,180,180,2266         [F92A]
741 DATA 180,224,224,128,0,0,0,0,128,1
    28,128,232,0,0,1372                      [E7E6]
742 DATA 0,0,0,0,0,0,232,232,232,232,232
    ,232,232,232,0,1856                      [F4CE]
743 DATA 0,0,0,0,0,0,192,152,152,152,1
    52,152,152,152,1256                      [16DC]
744 DATA 0,128,224,224,180,180,180,180,1
    28,128,128,128,0,0,0,1808                [6672]
745 DATA 0,0,0,0,0,0,0,0,152,152,152,1
    52,152,152,912                           [77D4]
746 DATA 152,152,180,180,180,180,180,180
    ,180,180,10,138,76,204,48,2220           [E872]
747 DATA 80,20,84,84,180,32,0,0,0,0,67
    ,0,0,0,547                               [657A]
748 DATA 0,0,0,0,10,138,12,204,16,80,2
    0,0,0,0,480                              [B6A8]
749 DATA 0,0,0,0,0,0,0,0,0,68,4,168,
    67,307                                   [FAD4]
750 DATA 0,0,0,0,0,69,10,142,68,0,0,0,
    0,40,329                                 [1E82]
751 DATA 67,5,0,0,0,138,4,0,0,138,8,68,0
    ,0,136,564                               [A890]
752 DATA 131,84,20,80,32,0,0,0,0,0,0,0
    ,0,0,347                                 [9474]
753 DATA 0,0,0,0,0,0,0,10,0,0,0,0,0,
    10                                       [54DE]
754 DATA 0,0,0,0,0,0,0,68,176,0,0,0,0,
    0,244                                    [2A48]
755 DATA 8,69,0,0,0,80,32,0,0,0,0,124,
    84,1,398                                 [90B6]
756 DATA 65,5,0,0,130,2,168,40,160,32,0,
    0,0,0,0,602                              [E7BA]
757 DATA 0,0,0,0,0,0,80,16,204,12,207,
    10,168,40,737                            [F98E]
758 DATA 0,0,0,0,8,152,0,0,0,0,0,0,0,0
    ,160                                     [1BD2]
759 DATA 0,0,120,16,0,0,0,0,40,84,0,0,0,
    0,0,260                                  [2D0A]
760 DATA 0,5,67,5,138,0,0,0,0,0,0,0,0,
    0,215                                    [1D4C]
761 DATA 68,0,0,0,1,124,160,32,80,8,138,
    0,0,0,0,611                              [2DC4]
762 DATA 0,0,0,0,0,0,0,0,0,124,0,0,69,
    4,197                                    [B650]
763 DATA 0,0,0,2,130,10,0,0,0,0,0,0,0,
    0,142                                    [F818]
764 DATA 0,0,0,0,112,0,0,0,0,0,0,2,188
    ,80,382                                  [2A16]
765 DATA 16,0,0,0,0,0,0,0,136,8,138,10
    ,138,0,446                               [E960]
766 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                             [5922]
767 DATA 0,0,20,80,16,68,4,65,2,168,0,0,
    0,0,0,423                                [5D14]
768 DATA 0,0,0,0,69,0,0,0,0,0,0,10,130
    ,0,209                                   [6FA4]
769 DATA 0,40,80,16,0,0,0,65,0,0,0.136
    ,76,138,551                              [D1E0]
770 DATA 10,0,0,0,0,80,32,0,0,0,135,2,40
    ,0,0,299                                 [617A]
771 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                             [541A]
772 DATA 138,5,65,139,5,130,2,0,0,0,32,3
    2,0,0,0,548                              [8BDA]
773 DATA 0,0,0,0,0,80,0,0,0,0,0,0,0,12
    4,204                                    [3528]
774 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                             [7B20]
775 DATA 0,0,84,20,80,16,76,2,0,0,0,0,0,
    0,0,278                                  [093C]
776 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                             [A124]
777 DATA 168,0,0,0,0,0,0,0,0,0,0,0,0,2
    0,188                                    [D04A]
778 DATA 80,32,138,10,130,2,168,0,0,0,16
    0,20,84,1,0,825                          [7370]
779 DATA 0,0,0,0,0,0,130,10,138,76,168
    ,1,65,0,588                              [9BE6]
780 DATA 0,0,0,0,0,0,10,138,4,68,16,0,
    0,0,236                                  [121C]
781 DATA 0,0,0,0,20,1,0,0,0,0,68,176,0,0
    ,0,265                                   [C5B4]
782 DATA 0,0,152,0,0,0,0,0,0,0,160,40,
    84,0,436                                 [1276]
783 DATA 0,0,0,0,0,135,0,0,0,0,0,142,0
    ,0,277                                   [B9A0]
784 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                             [5722]
785 DATA 169,0,80,32,136,8,138,10,130,40
    ,0,0,0,0,0,743                           [C708]
786 DATA 0,0,0,130,10,138,8,0,0,0,0,0,
    0,0,286                                  [5D18]
787 DATA 0,0,0,2,168,40,80,16,0,0,0,0,0,
    0,0,306                                  [7E22]
788 DATA 0,136,4,0,0,0,80,0,0,0,0,136,
    32,0,388                                 [1F9A]
789 DATA 0,0,0,0,0,17,51,34,51,51,17,0,0
    ,85,119,425                              [48E4]
790 DATA 119,17,34,51,51,17,170,187,187,
    51,51,51,51,34,34,1105                   [5062]
791 DATA 51,51,51,51,34,0,0,0,17,17,17,5
    1,34,0,0,374                             [8342]
792 DATA 0,0,34,34,34,51,34,5,0,0,17,17,
```

```
    17,51,17,311                                 [1B44]
793 DATA 10,0,34,34,34,51,17,0,0,0,0,81,
    81,243,243,828                               [0016]
794 DATA 243,243,81,243,243,81,81,81,81,
    162,81,0,0,0,17,1637                         [C49C]
795 DATA 243,243,243,85,255,85,51,34,243
    ,243,243,162,243,243,162,2778                [9150]
796 DATA 162,162,162,81,0,0,162,162,243,
    243,243,243,170,255,170,2458                 [39BA]
797 DATA 51,17,243,243,243,162,0,0,0,34,
    243,243,243,63,127,1912                      [19B2]
798 DATA 255,255,170,255,127,63,63,191,2
    55,85,255,255,191,63,63,2546                 [FE10]
799 DATA 63,0,0,63,63,63,63,63,63,0,0,63
    ,63,63,63,693                                [16F8]
800 DATA 42,42,42,42,42,42,42,42,63,63,6
    3,63,63,63,63,777                            [CB74]
801 DATA 63,63,63,63,63,63,63,63,63,21,2
    1,21,21,21,21,693                            [D958]
802 DATA 21,21,63,63,63,63,0,0,63,63,63,
    63,63,63,0,672                               [882E]
803 DATA 0,63,63,0,0,21,21,63,63,63,63,2
    1,42,42,63,588                               [9724]
804 DATA 63,63,63,63,81,81,243,243,243,2
    43,243,243,243,243,243,2601                  [BA54]
805 DATA 243,243,243,243,243,243,243,243
    ,243,243,243,243,243,162,162,3483            [17A8]
806 DATA 243,243,243,243,243,243,0,0,0,0
    ,0,162,243,243,0,2106                        [89AC]
807 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                                                 [AD1A]
808 DATA 0,0,0,0,0,81,243,243,0,81,81,24
    3,243,243,243,1701                           [AD90]
809 DATA 243,243,243,243,243,243,243,243
    ,243,243,243,243,243,243,243,3645            [53B0]
810 DATA 243,243,243,243,243,243,243,243
    ,243,243,0,0,0,0,0,2430                      [3E74]
811 DATA 81,81,243,0,0,0,81,243,243,243,
    243,0,81,243,243,2025                        [E8C8]
812 DATA 243,243,243,243,81,243,243,243,
    243,243,243,243,170,170,170,3264             [7D38]
813 DATA 170,170,255,119,255,0,0,0,0,0,0
    ,170,243,243,243,1868                        [A4CE]
814 DATA 243,243,243,243,243,243,0,162,1
    62,243,243,243,243,243,0,2997                [2716]
815 DATA 0,0,162,243,243,243,243,0,0,0,0
    ,0,162,162,243,1701                          [00DA]
816 DATA 128,128,128,128,192,128,128,192
    ,0,0,0,0,0,128,128,1408                      [46B4]
817 DATA 0,0,0,0,0,0,128,128,64,0,0,0,0,
    0,128,448                                    [54F2]
818 DATA 128,128,0,0,128,64,0,0,0,0,128,
    128,128,0,0,832                              [1A70]
819 DATA 0,0,0,192,128,128,128,192,128,1
    28,128,128,64,64,64,1472                     [C542]
820 DATA 128,0,0,0,0,64,128,128,192,12
    8,64,0,0,128,960                             [12DC]
821 DATA 64,64,128,64,128,64,64,192,64,6
    4,64,64,0,0,0,1024                           [A0D8]
822 DATA 128,0,0,0,128,0,0,64,64,64,64
    ,64,64,0,640                                 [DD4C]
823 DATA 0,0,192,64,0,0,0,128,128,128,12
    8,128,128,128,64,1216                        [DFE4]
824 DATA 64,64,64,64,64,64,64,128,64,64,
    64,64,64,64,64,1024                          [2874]
825 DATA 0,0,0,0,0,0,0,128,64,0,64,64,
    64,64,448                                    [A314]
826 DATA 64,64,0,0,0,0,0,0,0,0,0,64,12
    8,128,448                                    [9604]
827 DATA 128,128,64,0,0,128,0,0,0,0,128,
    128,128,128,192,1152                         [C976]
828 DATA 128,128,128,128,0,0,0,128,64,64
    ,64,64,51,51,51,1049                         [8A9E]
829 DATA 51,51,51,51,51,34,34,34,34,34,5
    1,51,51,51,51,680                            [A55C]
830 DATA 51,51,51,51,51,51,0,0,0,0,0,0
    ,0,0,306                                     [5B6C]
831 DATA 0,0,0,0,0,0,17,17,17,17,17,17
    ,17,17,136                                   [1B68]
832 DATA 0,0,0,0,0,0,0,51,51,51,51,51,
    51,17,323                                    [CFDE]
833 DATA 17,0,0,0,0,34,34,34,34,0,0,0,0,
    0,0,153                                      [A912]
834 DATA 0,0,243,243,243,243,162,81,243,
    243,243,243,162,162,162,2673                 [30A8]
835 DATA 0,243,243,0,0,0,0,0,243,243,0
    ,0,0,0,972                                   [FB48]
836 DATA 0,243,243,243,0,0,0,81,81,243,2
    43,243,81,81,243,2025                        [5CD6]
837 DATA 162,0,0,81,81,243,243,162,0,243
    ,243,243,162,0,81,1944                       [564A]
838 DATA 243,243,243,243,162,0,162,128,1
    62,81,0,0,0,0,243,1910                       [1A30]
839 DATA 243,162,0,0,0,0,0,162,243,0,0,0
    ,0,0,0,810                                   [E73E]
840 DATA 162,0,0,0,0,0,0,162,0,0,81,81
    ,81,243,810                                  [93B2]
841 DATA 243,243,243,243,243,243,243,243
    ,243,0,243,81,0,162,243,2916                 [40A4]
842 DATA 243,243,0,162,243,243,81,0,243,
    243,0,0,0,162,243,2106                       [771E]
843 DATA 81,243,243,0,0,0,0,0,162,243,
    243,243,226,0,1684                           [C792]
844 DATA 0,0,0,0,226,226,128,0,0,0,0,0,0
    ,0,0,580                                     [A774]
845 DATA 0,0,81,243,162,0,0,0,0,243,24
    3,0,0,0,972                                  [9ABC]
846 DATA 0,0,0,0,162,81,0,0,0,0,0,0,0,
    119,362                                      [2410]
847 DATA 119,119,119,119,119,17,17,243,1
    70,255,255,255,255,255,51,2368               [BABC]
848 DATA 243,243,243,243,81,81,247,85,24
    3,247,255,255,255,170,170,3061               [B5AC]
849 DATA 0,243,255,128,128,170,170,170,0
    ,243,243,0,0,0,0,1750                        [6BCA]
850 DATA 0,0,0,243,243,243,243,243,81,0,
    243,243,162,162,243,2349                     [7600]
851 DATA 243,243,243,0,0,0,0,0,162,0,0
    ,0,0,0,891                                   [2544]
852 DATA 0,0,0,243,17,17,17,17,17,51,51,
    119,51,51,51,702                             [90E0]
853 DATA 51,51,51,51,255,255,255,255,243
    ,243,243,243,243,251,255,2945                [4036]
854 DATA 255,255,243,243,243,243,255,247
    ,243,243,243,243,243,243,255,3697            [1BDE]
855 DATA 255,247,243,243,243,243,243,17,
    17,17,17,17,17,17,17,1853                    [92AE]
856 DATA 119,119,119,119,119,119,119,119
    ,17,17,21,21,63,63,63,1217                   [1B0A]
857 DATA 63,55,127,127,63,63,63,63,63,17
    ,17,55,55,55,55,941                          [6178]
858 DATA 17,85,170,119,119,119,119,119,2
    55,255,0,0,0,234,191,1802                    [BF9C]
859 DATA 191,191,191,0,64,149,63,63,123,
    183,183,0,128,106,63,1698                    [7BC0]
860 DATA 63,63,123,63,0,64,149,63,63,183
    ,183,183,63,63,63,1389                       [0896]
861 DATA 63,63,63,192,0,183,183,123,63,1
    06,128,0,0,63,63,1293                        [10FC]
862 DATA 63,63,149,64,0,0,183,183,183,63
    ,106,128,0,0,0,1185                          [4C2C]
863 DATA 0,128,106,63,243,183,183,0,0,64
    ,149,63,243,123,123,1671                     [C320]
864 DATA 123,63,243,63,63,149,64,0,243,1
    23,243,63,63,106,128,1737                    [C0A0]
865 DATA 0,0,192,63,63,63,123,123,123,0,
    128,106,63,63,243,1353                       [4846]
866 DATA 63,63,0,0,0,192,63,183,183,183,
    0,64,149,63,63,1269                          [9F52]
867 DATA 243,183,183,183,183,183,63,63,6
    3,192,0,243,63,243,63,2151                   [1F18]
868 DATA 106,128,0,0,123,123,123,63,192,
    128,0,0,243,63,243,1535                      [C7A2]
869 DATA 63,149,64,0,0,0,192,63,63,63,24
    3,63,63,0,0,1026                             [DB02]
870 DATA 128,106,106,106,106,106,243,123
    ,243,63,192,0,0,0,106,1628                   [61C0]
871 DATA 106,106,106,106,128,0,0,17,17,1
    7,17,17,17,17,17,688                         [5196]
872 DATA 255,119,119,119,119,119,119,119
    ,0,0,0,0,0,0,0,1088                          [8812]
```

Listing 3. Der Basic-Lader erzeugt das Titelbild (Schluß)

```
10 MEMORY &7FFF                                  [8052]
20 FOR ad=&8000 TO &8050                         [0F9E]
30 READ i:POKE ad,i                              [C216]
40 NEXT                                          [B986]
50 SAVE "prscreen.bin",b,&8000,&51               [0282]
60 END                                           [0EBA]
70 DATA &f3,&dd,&5e,&00,&dd,&56,&01,&dd          [F5B8]
80 DATA &6e,&02,&dd,&66,&03,&dd,&21,&00          [8A8A]
90 DATA &c0,&06,&19,&c5,&06,&28,&c5,&dd          [66A4]
100 DATA &e5,&4e,&06,&00,&c5,&fd,&e1,&d5
                                                 [AAB2]
110 DATA &fd,&29,&fd,&29,&fd,&29,&fd,&29
                                                 [E448]
120 DATA &fd,&19,&11,&00,&08,&06,&08,&fd
                                                 [3C8E]
130 DATA &7e,&00,&dd,&77,&00,&fd,&7e,&08
                                                 [D864]
140 DATA &dd,&77,&01,&dd,&19,&fd,&23,&10
                                                 [B058]
150 DATA &ee,&d1,&dd,&e1,&dd,&23,&dd,&23
                                                 [8ED6]
160 DATA &23,&c1,&10,&ca,&c1,&10,&c4,&fb
                                                 [DF8E]
170 DATA &c9                                     [1A68]
```

Listing 4. Laderoutine für erzeugte Grafiken

Sag' mir, wo die Sonne steht

Wenn Sie beim ersten Sonnenstrahl aus den Federn wollen, müssen Sie sich einen Wecker stellen. Wann die Sonne aufgeht, sagt Ihnen Ihr Computer auf die Minute genau, und das für jeden beliebigen Ort der Erde. Auf Wunsch erhalten Sie eine gedruckte Übersicht.

Dieses Programm wendet sich an alle, die es interessiert, wann an ihrem Heimatort – oder vielleicht auch in Japan – die Sonne oder der Mond auf- oder untergeht. Zusätzlich gibt es Auskunft über die Höhe dieser beiden Gestirne über und unter dem Horizont während eines bestimmten Tages an jedem Ort der Erde.

Außer bei den Eingabedaten für den Ort und das Datum sind bei allen Eingaben nur Kennziffern oder Buchstaben ohne Drücken der Enter-Taste notwendig.

Nach dem Start erscheint zunächst ein Titelbild und nach Betätigung irgendeiner Taste das Hauptmenü. Zu allen Menüpunkten sind hier auch jeweils Erläuterungen auf den Bildschirm zu rufen, was dem Bedienungskomfort dient.

Bei der Anwahl eines Menüpunktes, dessen Programmteil Berechnungen ausführt, verzweigt der Computer in ein Untermenü. Dort können Sie dann die vorgegebenen Daten für den Bezugsort ändern. Das geschieht durch die Eingabe des Ortes, seiner geografischen Breite und Länge, sowie der Bezugslänge für die Zeitzone. Hier geben Sie auch das jeweilige Datum ein. Dazu stehen zwei verschiedene Möglichkeiten zur Auswahl. Entweder wird nach Monat und Jahr oder aber nach dem Datum gefragt:

Monat und Jahr: MM.JJ (zum Beispiel »10.86« oder »2.86«, nicht aber »2.1986« oder »Feb.86«)

Datum: TT.MM.JJ (zum Beispiel »13.02.86«, nicht aber 13.2.86 oder 13.02.1986)

Wurde im Hauptmenü die Kennziffer 1 (Tabelle der Sonnen- und Monddaten) oder 3 (Tagesgang der Sonnen- und Mondhöhe) gewählt, so werden während der Berechnung der Sonnen- und Monddaten für jede halbe Stunde des betreffenden Tages die wichtigsten Daten für Sonne und Mond in tabellarischer Form ausgegeben:

- Kulmination (Höchststand des Gestirns)
- Beginn der Dämmerung (die Sonne steht noch 18 Grad unter dem Horizont)
- Ende der Dämmerung (die Sonne steht 18 Grad unter dem Horizont)
  Auf- und Untergang (Höhe des Gestirns 0 Grad)
- Mondphase (0 % = Neumond, 50 % = Halbmond, zunehmend, 100 % = Vollmond, -50 % = Halbmond, abnehmend)
- Horizontalparallaxe (Winkel, unter dem der Erdradius vom Mond aus gesehen wird)
- Durchmesser des Mondes im Winkelmaß, stündliche Bewegung in Länge und Breite am Himmel

Nach dem Drücken irgendeiner Taste erscheint dann je nach gewählter Kennziffer (1 oder 3) die Tabelle oder der Tagesgang.

Die Tabelle in Bild 1 zeigt den Ausschnitt eines typischen Beispiels.

Jülich 13.02.86						
	Sonne			Mond		
Zeit	Rekt.	Dekl.	Höhe	Rekt.	Dekl.	Höhe
0	21 45	-13.52	-51.2	0 48	3.28	-16.6
1	21 45	-13.50	-50.5	0 50	3.52	-23.9
2	21 45	-13.48	-49.9	0 52	3.76	-29.8
und so weiter						
Sonne: Kulmination.............						
Mond:						

Bild 1. Tabelle der Sonnen- und Monddaten

- »Rekt.« steht für Rektaszension = Länge am Himmel (in Stunden und Minuten)
- »Dekl.« ist die Abkürzung für Deklination = Breite am Himmel (in Grad. Bogenminuten)
- Höhe = geometrische Höhe in Grad über (+) oder unter (–) dem Horizont

Nach erneuter Betätigung einer Taste fragt der Computer nach einem neuen Datum. Verneint man diese Frage (einfach »N« eingeben), können Sie sich mit »J« den Tagesgang von Sonne und Mond ausgeben lassen.

Die Grafik verdeutlicht den Tagesgang der betreffenden Gestirne. Der Verlauf der Höhe über oder unter dem Horizont wird gegen die Tageszeit aufgetragen. Die Abszisse der Grafik steht also für die Tageszeit, die Zahlen für die genaue Uhrzeit. An der Ordinate sind die Höhen der Gestirne Sonne und Mond abzulesen. Die Zahlen sind in Grad angegeben. Oberhalb der Grafik erscheinen der gewählte Ort und das jeweilige Datum sowie Sonnenaufgang, Kulmination und Sonnenuntergang. Unterhalb der Grafik sind Mondphase sowie Auf- und Untergang aufgeführt.

Wählt man im Hauptmenü die Kennziffer 5 oder 7 (Ausgabe einer Monatstabelle am Bildschirm oder am Drucker), erhält man beispielsweise das Resultat aus Bild 2.

(Hans-Werner Pätz/ja)

Jülich	Breite = 51.0 Grad		Länge = -6.3 Grad	
Datum	Dämm.	Aufg.	Unterg.	Dämm.
	h m	h m	h m	h m
1.2.86	6 56	8 18	17 19	18 40
2.2.86	6 55	8 16	17 21	18 42
3.2.86	....			
Kulmination		Mondphase		
h	m	Grad	in %	
12	48	21.9	–45.2	
12	48	22.2	–52.4	

Bild 2. Monatstabelle für Sonne und Mond

Programm-Steckbrief	
Name:	Sonne und Mond
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
10 REM ####################################
   ##                                         [259C]
20 REM            COPYRIGHT                   [193E]
30 REM          FEBRUAR 1986                  [C44C]
40 REM       HANS-WERNER PAETZ                [466C]
50 REM         5170 JUELICH                   [13F4]
60 REM ####################################
   ##                                         [AEA6]
70 DEG                                        [D6AE]
80 INK 0,0                                    [462C]
90 INK 1,26                                   [9BA0]
100 INK 2,18                                  [9FF4]
110 INK 3,9                                   [2598]
120 DIM SOHOEHE(48),MONDHOEHE(48),MOPHAS
    E(48)                                     [2FD6]
130 DIM SOREKT(48),SODEKL(48),MOREKT(48)
    ,MODEKL(48)                               [669C]
140 DSO=0.9856473 : REM TAEGLICHE BEWEGU
    NG DER SONNE                              [A58C]
150 DMO=13.1763965 : REM TAEGL.BEWEGUNG
    DES MONDES                                [1352]
160 DMP=0.1114041  : REM TAEGL.BEWEGUNG
    DES MONDPERIGAEUMS                        [16B0]
170 DMK=0.0529539  : REM TAEGL.BEWEGUNG
    DES MONDKNOTENS                           [BF32]
180 SOL0=335.73 :    REM BEZUGSLAENGE DER
     SONNE                                    [CB3E]
190 SOP0=282.55 :    REM BEZUGSPERIHEL DE
    R SONNE                                   [F704]
200 MOL0=67.88 :     REM BEZUGSLAENGE DES
     MONDES                                   [0E9C]
210 MOP0=233.75 :    REM BEZUGSPERIHEL DE
    S MONDES                                  [BD6E]
220 MOK0=206.92 :    REM BEZUGSLAENGE DES
     AUFSTEIGENDEN KNOTEN                     [CE80]
230 JUL0=2443200.5: REM BEZUGS-JUL.DATUM
                                              [3A7C]
240 EKL =23.44 :     REM EKLIPTIK             [174A]
250 DEF FNM(A,B)=B*(A/B-INT(A/B))             [829C]
260 DEF FNASIN(X)=ATN(X/SQR(1-X*X))           [DA0C]
270 DEF FNACOS(X)=ATN(SQR(1-X*X)/X)           [D904]
280 SYMBOL AFTER 237                          [B7BC]
290 SYMBOL 237,14,15,7,231,119,57,31,14       [C82A]
300 SYMBOL 238,112,240,224,231,238,156,2
    48,48                                     [0D82]
310 SYMBOL 239,14,14,14,15,30,60,120,240
                                              [4C5E]
320 SYMBOL 240,48,240,240,240,120,60,30,
    15                                        [D922]
330 BORDER 12,18                              [3918]
340 MODE 1                                    [C05A]
350 LOCATE 8,6: PRINT "PROGRAMM ZUR BERE
    CHNUNG DER"                               [F0EE]
360 LOCATE 10,7:PRINT "SONNEN- UND MONDH
    OEHE"                                     [3C78]
370 PRINT STRING$(40,"_")                     [1BCC]
380 LOCATE 15,11:PRINT "COPYRIGHT"            [F4D6]
390 LOCATE 10,12:PRINT "HANS-WERNER PAET
    Z"                                        [68FA]
400 LOCATE 12,13:PRINT "5170 JUELICH"         [5F36]
410 LOCATE 17,14:PRINT "1986"                 [D312]
420 PRINT STRING$(40,"_")                     [1EC4]
430 LOCATE 18,18:PRINT CHR$(237);CHR$(23
    8)                                        [A318]
440 LOCATE 18,19:PRINT CHR$(239);CHR$(24
    0)                                        [1312]
450 LOCATE 6,23:PRINT "Fortfahren im Pro
    gramm durch"                              [4446]
460 LOCATE 4,24:PRINT "DRUECKEN einer be
    liebigen Taste"                           [BB9C]
470 CALL &BB18                                [7414]
480 BORDER 12                                 [30FA]
490 MODE 1                                    [C366]
500 LOCATE 8,2:PRINT "WELCHER PROGRAMMTE
    IL"                                       [8916]
510 LOCATE 2,5:PRINT "ALLGEMEINE ERLAEUT
    ERUNGEN .......... 0"                     [9D42]
520 LOCATE 2,7:PRINT "TABELLE DER SONNEN
    - UND MONDDATEN .. 1"                     [E9DE]
530 LOCATE 2,8:PRINT "ERKLAERUNGEN ZUR T
    ABELLE ........... 2"                     [28F4]
540 LOCATE 2,10:PRINT "TAGESGANG DER SON
    NEN- UND MONDHOEHE{2 SPACE}3"             [4BAC]
550 LOCATE 2,11:PRINT "ERKLAERUNGEN ZUM
    TAGESGANG ......... 4"                    [EBBE]
560 LOCATE 2,13:PRINT "MONATSTABELLE FUE
    R DIE SONNE........ 5"                    [27EE]
570 LOCATE 2,14:PRINT "ERKLAERUNGEN ZUR
    MONATSTABELLE ..... 6"                    [8ADA]
580 LOCATE 2,15:PRINT "AUSGABE DER TABEL
    LE AM DRUCKER...... 7"                    [16B2]
590 LOCATE 2,17:PRINT "E N D E .........
    ................... 9"                    [1A08]
600 LOCATE 3,20:PRINT "KENNZIFFER ---> "
    ;                                         [7558]
610 EIN$=INKEY$:IF EIN$="" THEN 610 ELSE
     PRINT EIN$                               [243C]
620 KENN=VAL(EIN$)                            [BEAA]
630 IF KENN=0 THEN 2780                       [F782]
640 IF KENN=9 THEN 2740                       [FE8E]
650 IF KENN=1 OR KENN=3 OR KENN=5 OR KEN
    N=7 THEN 680                              [9F1C]
660 ON KENN/2 GOTO 3050,3180,2900             [C46C]
670 GOTO 480                                  [7F64]
680 BEZLAE=-15:ORT$="JUELICH":BREITE=51:
    LAENGE=-6.33                              [D496]
690 MODE 1:LOCATE 14,5:PRINT "EINGABEDAT
    EN"                                       [EA1E]
700 PRINT STRING$(&50,"_")                    [D514]
710 LOCATE 2,11:PRINT "ORT:{5 SPACE}";OR
    T$                                        [783C]
720 LOCATE 2,13:PRINT "BREITE:{2 SPACE}"
    ;:PRINT USING "##.#";BREITE;:PRINT "
     GRAD"                                    [B4E0]
730 LOCATE 2,14:PRINT "LAENGE: ";:PRINT
    USING "###.#";LAENGE;:PRINT " GRAD"       [4BAE]
740 LOCATE 2,16:PRINT "BEZUGSLAENGE FUER
     DIE ZEITZONE:{2 SPACE}";BEZLAE " GR
    AD":LOCATE 2,17:PRINT "{4 SPACE}(z.B
    . fuer MEZ: -15{2 SPACE}) "               [14F8]
750 LOCATE 2,19:PRINT "- oestl.{3 SPACE}
    + westl. vom Nullmeridian"                [B94E]
760 LOCATE 2,25:PRINT SPACE$(38):LOCATE
    2,25:PRINT "A E N D E R N{3 SPACE}(J
    /N)"                                      [4688]
770 EIN$=INKEY$ : IF EIN$="" THEN 770         [DEAC]
780 IF UPPER$(EIN$)<>"J" THEN 840             [1704]
790 LOCATE 1,25:PRINT SPACE$(38):LOCATE
    2,25:INPUT "WELCHER ORT ";ORT$            [FA4E]
800 LOCATE 1,25:PRINT SPACE$(38):LOCATE
    2,25:INPUT "GEOGR.BREITE ";BREITE         [ADFE]
810 LOCATE 1,25:PRINT SPACE$(38):LOCATE
    2,25:INPUT "LAENGE ";LAENGE               [1380]
820 LOCATE 2,25:PRINT SPACE$(38):LOCATE
    2,25:INPUT "BEZUGSLAENGE FUER DIE ZE
    ITZONE ";BEZLAE                           [17EA]
830 GOTO 690                                  [EA66]
840 IF KENN=5 OR KENN=7 THEN LOCATE 1,25
    :PRINT SPACE$(38):LOCATE 2,25:INPUT
    "WELCHER MONAT UND JAHR ";DATUM$:DAT
    UM$="01."+DATUM$                          [47DE]
850 IF KENN=5 OR KENN=7 THEN MODE 2:WIND
    OW #1,1,80,1,3:WINDOW #2,1,80,4,23:G
    OTO 880                                   [DC80]
860 LOCATE 1,25:PRINT SPACE$(38):LOCATE
    2,25:INPUT "WELCHES DATUM (z.B.08.07
    .84)";DATUM$                              [6364]
870 IF LEN(DATUM$)<8 THEN 860                 [6C7C]
880 JAHR=VAL(RIGHT$(DATUM$,2))                [698C]
890 MONTH = VAL(MID$(DATUM$,4,2))             [1A48]
900 IF KENN=7 THEN PRINT #8:PRINT #8:PRI
    NT #8,ORT$;SPACE$(15);"BREITE = ";:P
    RINT #8,USING "##.#";BREITE;:PRINT #
    8," GRAD{8 SPACE}LAENGE = ";:PRINT #
    8,USING "###.#";LAENGE;:PRINT #8," G
    RAD"                                      [C3A8]
910 IF KENN=7 THEN PRINT #8:PRINT #8,"{2
     SPACE}DATUM{4 SPACE}DAEMM. AUFG.{2
    SPACE}UNTERG. DAEMM.{7 SPACE}KULMINA
    TION{4 SPACE}MONDPHASE"                   [DE5E]
915 IF KENN=7 THEN PRINT #8,         "{1
    1 SPACE}h{2 SPACE}m{3 SPACE}h{2 SPAC
    E}m{5 SPACE}h{2 SPACE}m{3 SPACE}h{2
    SPACE}m{8 SPACE}h{2 SPACE}m{4 SPACE}
    Grad{6 SPACE}in %"                        [0730]
920 IF KENN=5 THEN LOCATE #1,2,1:PRINT #
    1,ORT$:LOCATE #1,30,1:PRINT #1,"BREI
    TE = ";:PRINT #1,USING "##.#";BREITE
    ;:PRINT #1," GRAD{8 SPACE}LAENGE = "
    ;:PRINT #1,USING "###.#";LAENGE;:PRI
    NT #1," GRAD"                             [E394]
930 IF KENN=5 THEN PRINT #1:PRINT #1,"{2
     SPACE}DATUM{4 SPACE}DAEMM. AUFG.{2
    SPACE}UNTERG. DAEMM.{7 SPACE}KULMINA
    TION{9 SPACE}MONDPHASE"                   [5F82]
940 DAY = VAL(LEFT$(DATUM$,2))                [BAD2]
950 IF MONTH>2 THEN 970                       [24E4]
960 TAGZAHL=365*JAHR+DAY+31*(MONTH-1)+IN
    T((JAHR-1)/4) : GOTO 980                  [6AFA]
970 TAGZAHL=365*JAHR+DAY+31*(MONTH-1)-IN
    T(0.4*MONTH+2.3)+INT(JAHR/4)              [8A88]
980 JULDAT0UT=TAGZAHL+2415019.5               [3834]
990 JULDAT0OZ=JULDAT0UT-(BEZLAE-LAENGE)/
    15/24                                     [36DA]
1000 STERNZEIT0UT=FNM(155.72835+(JULDAT0
     UT-JUL0)*DSO,360)                        [96F2]
1010 JULDELTA=JULDAT0OZ-JUL0+0.5 : REM J
     UL.DATUM UM 12 UHR OZ                    [738C]
1020 SONNENLAENGEM=FNM(SOL0+DSO*JULDELTA
     ,360)                                    [A21E]
1030 IF SONNENLAENGEM<0 THEN SONNENLAENG
     EM=SONNENLAENGEM+360                     [8E9A]
1040 SONNENLAENGE=FNM(SONNENLAENGEM+1.92
     *SIN(SONNENLAENGEM-SOP0),360)            [BEA6]
```

Listing. Sonne, Mond und Sterne

```
1050 IF SONNENLAENGE<0 THEN SONNENLAENGE
     =SONNENLAENGE+360                          [0BD0]
1060 XDEKL=SIN(EKL)*SIN(SONNENLAENGE)          [F92A]
1070 IF XDEKL=0 THEN XDEKL=0.000000001         [BF5E]
1080 IF XDEKL=1 THEN XDEKL=1-0.000000001
                                                [CC1E]
1090 SODEKL=FNASIN(XDEKL)                       [80E2]
1100 XREKT=COS(SONNENLAENGE)/COS(SODEKL)
                                                [090E]
1110 IF XREKT=0 THEN XREKT=0.000000001         [53AC]
1120 IF XREKT=1 THEN XREKT=1-0.000000001
                                                [1C6C]
1130 SOREKT=FNACOS(XREKT)                       [E126]
1140 IF SOREKT < 0 THEN SOREKT=SOREKT+18
     0                                          [ED72]
1150 IF SODEKL<0 THEN SOREKT=360-SOREKT        [B5CC]
1160 MOLM=FNM(MOL0+DMO*JULDELTA,360)           [94E6]
1170 MOAM=FNM(MOLM-(MOP0+DMP*JULDELTA),3
     60)                                        [BB42]
1180 MOK=FNM(MOK0-DMK*JULDELTA-0.15*SIN(
     SONNENLAENGEM-SOP0),360)                   [7E26]
1190 EVEK=FNM(1.3*SIN(2*MOLM-2*SONNENLAE
     NGE-MOAM),360)                             [CE5E]
1200 ANEQ=FNM(-0.19*SIN(SONNENLAENGEM-SO
     P0),360)                                   [505E]
1210 MOAK=MOAM+EVEK+ANEQ-0.37*SIN(SONNEN
     LAENGEM-SOP0)                              [FC18]
1220 MOLK=MOLM+EVEK+ANEQ+6.29*SIN(MOAK)        [4C6E]
1230 MOL2=MOLK+0.6*SIN(2*MOLM-2*SONNENLA
     ENGE)                                      [4B56]
1240 MOLAENGE=FNM(MOL2-0.12*SIN(2*MOL2-2
     *MOK),360)                                 [846C]
1250 MOBREITE=5.15*SIN(MOL2-MOK)+0.15*SI
     N(MOL2-2*SONNENLAENGE+MOK)                 [8DBC]
1260 PHASE=MOLAENGE-SONNENLAENGE               [5212]
1270 IF PHASE<0 THEN PHASE=PHASE+360           [8190]
1280 IF PHASE<=180 THEN MOPHASE=PHASE/1.
     8 ELSE MOPHASE=(180-PHASE)/1.8             [D71C]
1290 H=0.954+0.056*COS(MOAK)   : REM HORI
     ZONTALPARALLAXE                            [7ABE]
1300 D=120*(0.29*H-0.01)       : REM MOND
     DURCHMESSER IN BOGENMINUTEN                [661A]
1310 NMOL=0.55+0.06*COS(MOAK) : REM STDL
     .BEWEG. IN MONDLAENGE                      [FD46]
1320 NMOB=0.05*COS(MOL2-MOK)   : REM STDL
     .BEWEG. IN MONDLAENGE                      [D398]
1330 REM BERECHNUNG VON KULMINATION, AUF
     - UND UNTERGANG DER SONNE                  [7F1E]
1340 SOLA=SONNENLAENGEM-SONNENLAENGE : S
     OL=SONNENLAENGE                            [8ED4]
1350 IF SOLA>180 THEN SOLA=SOLA-360            [18DC]
1360 ZEITGLEICH=(SOLA+(SOL-SOREKT))/15*6
     0:REM ZEITGLEICHUNG IN MINUTEN             [3054]
1370 KULMINATION=12-ZEITGLEICH/60+(LAENG
     E-BEZLAE)/15                               [D13A]
1380 X=-TAN(SODEKL)*TAN(BREITE)                [BD7A]
1390 HATABO=FNACOS(X) : IF HATABO<0 THEN
      HATABO=HATABO+180                         [0F9C]
1400 HATABO=HATABO/15                           [122A]
1410 KULHOEHE=90-BREITE+SODEKL                 [6E6C]
1420 IF BREITE<0 THEN KULHOEHE=180-(90-B
     REITE+SODEKL)                              [0D26]
1430 AUFGANG=KULMINATION-HATABO                [7D8A]
1440 UNTERGANG=KULMINATION+HATABO              [B9EC]
1450 SOKUL=INT(KULMINATION)+(KULMINATION
     -INT(KULMINATION))*0.6                     [8BEA]
1460 SOAUF=INT(AUFGANG)+(AUFGANG-INT(AUF
     GANG))*0.6                                 [A9E0]
1470 SOUNTER=INT(UNTERGANG)+(UNTERGANG-I
     NT(UNTERGANG))*0.6                         [FA72]
1480 COSDAEM=(-0.207912-SIN(BREITE)*SIN(
     SODEKL))/(COS(BREITE)*COS(SODEKL))         [5928]
1490 DAEM=FNACOS(COSDAEM) : IF DAEM<0 TH
     EN DAEM=DAEM+180                           [3D26]
1500 DAEM=DAEM/15                               [24CC]
1510 DAEMAUF=KULMINATION-DAEM                  [0E4C]
1520 DAEMUNTER=KULMINATION+DAEM                [7BAE]
1530 AUFDAEM=INT(DAEMAUF)+(DAEMAUF-INT(D
     AEMAUF))*0.6                               [D1A2]
1540 UNTDAEM=INT(DAEMUNTER)+(DAEMUNTER-I
     NT(DAEMUNTER))*0.6                         [DF06]
1550 IF KENN=5 THEN 2530                        [12E2]
1560 IF KENN=7 THEN 2640                        [68EC]
1570 MODE 2                                     [08C8]
1575 LOCATE 1,25:PRINT SPACE$(10);" B i
     t t e{3 SPACE}w a r t e n (es wird
     noch gerechnet)"                           [D136]
1580 LOCATE 15,3:PRINT "SONNEN- UND MOND
     DATEN FUER ";ORT$;" AM ";DATUM$            [58C4]
1590 LOCATE 5,6:PRINT "S{2 SPACE}O{2 SPA
     CE}N{2 SPACE}N{2 SPACE}E"                  [EF18]
1600 LOCATE 5,8:PRINT "KULMINATION: ";:P
     RINT USING "##.##";SOKUL;:PRINT " U
     HR IN ";:PRINT USING "##.#";KULHOEH
     E;:PRINT " GRAD HOEHE"                     [A1C6]
1610 LOCATE 5,12:PRINT "AUFGANG: ";:PRIN
     T USING "##.##";SOAUF;:PRINT " UHR{
     21 SPACE}UNTERGANG: ";:PRINT USING
     "##.##";SOUNTER;:PRINT " UHR"              [4482]
1620 LOCATE 5,10:PRINT "Anfang der Daemm
     erung: ";:PRINT USING "##.##";AUFDA
     EM;:PRINT " Uhr{7 SPACE}Ende der Da
     emmerung: ";:PRINT USING "##.##";UN
     TDAEM;:PRINT " Uhr"                        [A0EC]
1640 LOCATE 5,15:PRINT "M{2 SPACE}O{2 SP
     ACE}N{2 SPACE}D"                           [0146]
1650 LOCATE 5,17:PRINT "MONDPHASE: ";:PR
     INT USING "###.#";MOPHASE;:PRINT "
     %"                                         [78D4]
1660 LOCATE 5,21:PRINT "HORIZONT.PARALL.
     :{2 SPACE}";:PRINT USING "##.##";H;
     :PRINT " GRAD{4 SPACE}DURCHMESSER:{
     7 SPACE}";:PRINT USING "##.#";D;:PR
     INT " MINUTEN"                             [5AAE]
1670 LOCATE 5,22:PRINT "STDL.BEW.IN LAEN
     GE:";:PRINT USING "##.##";NMOL;:PRI
     NT " GRAD{4 SPACE}STDL.BEW.IN BREIT
     E:";:PRINT USING "##.##";NMOB;:PRIN
     T " GRAD"                                  [011A]
1680 FOR I=0 TO 48                              [5DF6]
1690 ZEIT = I/2                                 [A8A6]
1700 UZEIT = ZEIT - (LAENGE-BEZLAE)/15         [5122]
1710 STERNZEITOZ=FNM(STERNZEIT0UT+UZEIT*
     15*1.00273791+0.002737909*LAENGE*15
     ,360)                                      [E4CA]
1720 SOL=SONNENLAENGE+(ZEIT-12)*0.041069
                                                [302C]
1730 SODEKL(I)=FNASIN(SIN(EKL)*SIN(SOL))
                                                [88FC]
1740 SOREKT(I)=FNACOS(COS(SOL)/COS(SODEK
     L(I)))                                     [EF16]
1750 IF SOREKT(I)<0 THEN SOREKT(I)=SOREK
     T(I)+180                                   [7A9C]
1760 IF SODEKL(I)<0 THEN SOREKT(I)=360-S
     OREKT(I)                                   [3B76]
1770 TZEIT = STERNZEITOZ-SOREKT(I)             [7B18]
1780 IF TZEIT < 0 THEN TZEIT=TZEIT +360        [E316]
1790 SINH=SIN(BREITE)*SIN(SODEKL(I))+COS
     (BREITE)*COS(SODEKL(I))*COS(TZEIT)         [D7BE]
1800 SOHOEHE(I)=FNASIN(SINH)                   [5A1A]
1810 MOL=MOLAENGE+(ZEIT-12)*NMOL               [15FE]
1820 MOB=MOBREITE+(ZEIT-12)*NMOB               [8AF6]
1830 SINDEKL=SIN(MOB)*COS(EKL)+COS(MOB)*
     SIN(EKL)*SIN(MOL)                          [4AAA]
1840 MODEKL(I)=FNASIN(SINDEKL)                 [3B34]
1850 REKT1=COS(MOB)*COS(MOL)                   [4ADC]
1860 REKT2=COS(MOB)*COS(EKL)*SIN(MOL)-SI
     N(MOB)*SIN(EKL)                            [BA70]
1870 MOREKT(I)=ATN(REKT2/REKT1)                [B096]
1880 IF REKT1 >=0 AND REKT2>=0 THEN 1920
                                                [5066]
1890 IF REKT1 >=0 AND REKT2 <0 THEN MORE
     KT=MOREKT+360 : GOTO 1920                  [551A]
1900 MOREKT(I)=MOREKT(I)+180                   [D386]
1910 IF MOREKT(I)<0 THEN MOREKT(I)=MOREK
     T(I)+360                                   [7E74]
1920 IF MOREKT(I)<0 THEN MOREKT(I)=MOREK
     T(I)+360                                   [8776]
1930 TZEITM=STERNZEITOZ-MOREKT(I)              [C422]
1940 SINHM=SIN(BREITE)*SIN(MODEKL(I))+CO
     S(BREITE)*COS(MODEKL(I))*COS(TZEITM
     )                                          [D4D4]
1950 MONDHOEHE(I)=FNASIN(SINHM)                [A0D8]
1960 IJ=I-1                                     [C4CE]
1970 IF I=0 THEN 2050                           [1418]
1980 IF MONDHOEHE(I)/MONDHOEHE(IJ) >= 0
     THEN 2050                                  [6B3A]
1990 DELTAHOEHE=ABS(MONDHOEHE(IJ))+ABS(M
     ONDHOEHE(I))                               [3950]
2000 DELTAZEIT =ABS(MONDHOEHE(IJ))*0.5/D
     ELTAHOEHE                                  [768C]
2010 ZEIT=IJ/2+DELTAZEIT                        [3642]
2020 UHR =INT(ZEIT)+0.6*(ZEIT-INT(ZEIT))
                                                [BD86]
2030 IF MONDHOEHE(IJ)>MONDHOEHE(I) THEN
     LOCATE 32,19:PRINT "UNTERGANG: ";:P
     RINT USING "##.##";UHR;:PRINT " UHR
     ":MOUNTER=UHR:GOTO 2050                    [A97E]
2040 LOCATE 5,19:PRINT "AUFGANG: ";:PRIN
     T USING "##.##";UHR;:PRINT " UHR":M
     OAUF=UHR                                   [D194]
2050 NEXT                                       [DD4C]
2060 LOCATE 1,25:PRINT SPACE$(10);"Fortf
     ahren im Programm durch Druecken ir
     gendeiner Taste"                           [61D4]
2070 CALL &BB18                                 [C470]
2080 IF KENN=3 THEN 2300                        [75D2]
2090 IF KENN=5 THEN 2530                        [89E2]
2100 IF KENN=7 THEN 2640                        [DDDA]
2110 MODE 2                                     [40B6]
2120 LOCATE 1,1:PRINT ORT$;:LOCATE 70,1:
     PRINT DATUM$                               [6CC4]
2130 LOCATE 17,2:PRINT "{3 SPACE}S O N N
     E{26 SPACE}M O N D"                        [58C0]
```

Listing. Sonne, Mond und Sterne (Fortsetzung)


HAPPY COMPUTER
WWW.HOMECOMPUTERWORLD.COM

```
2140 LOCATE 1,3:PRINT "ZEIT{8 SPACE}REKT
     .{5 SPACE}DEKL.{4 SPACE}HOEHE{10 SP
     ACE}REKT.{5 SPACE}DEKL.{4 SPACE}HOE
     HE"                                    [A3D8]
2150 WINDOW #1,1,80,4,23                    [232C]
2160 FOR I=0 TO 48 STEP 2                   [2046]
2170 PRINT #1, USING "###";I/2,:PRINT #1
     ,SPACE$(8);:PRINT #1,USING "##";INT
     (SOREKT(I)/15);:PRINT #1,USING "###
     #";INT(60*(SOREKT(I)/15-INT(SOREKT(
     I)/15)));                              [5D1A]
2180 PRINT #1,SPACE$(3);:PRINT #1,USING
     "###.##";SODEKL(I),:PRINT #1,SPACE$
     (4);:PRINT #1,USING "###.#";SOHOEHE
     (I);:PRINT #1,SPACE$(8);               [0DE0]
2190 PRINT #1,USING "####";INT(MOREKT(I)
     /15),:PRINT #1,USING "####";INT(60*
     (MOREKT(I)/15-INT(MOREKT(I)/15)));:
     PRINT #1,SPACE$(3);                    [EA88]
2200 PRINT #1,USING "###.##";MODEKL(I),:
     PRINT #1,SPACE$(4);:PRINT #1,USING
     "###.#";MONDHOEHE(I)                   [0A86]
2210 NEXT I                                 [6D1A]
2220 LOCATE 1,24:PRINT "KULMINATION UM "
     ;:PRINT USING "##.##";SOKUL;:PRINT
     " UHR{3 SPACE}AUFGANG: ";:PRINT USI
     NG "##.##";SOAUF;:PRINT " UHR{5 SPA
     CE}UNTERGANG: ";:PRINT USING "##.##
     ";SOUNTER;:PRINT " UHR"                [A69E]
2230 PRINT "M O N D : PHASE =";:PRINT US
     ING "###.#";MOPHASE;:PRINT "%{4 SPA
     CE}AUFGANG: ";:PRINT USING "##.##";
     MOAUF;:PRINT " UHR{5 SPACE}UNTERGAN
     G: ";:PRINT USING "##.##";MOUNTER;:
     PRINT " UHR"                           [389A]
2240 CALL &BB18                             [FC6E]
2250 LOCATE 1,25:PRINT SPACE$(78):LOCATE
      1,25:PRINT "ANDERES DATUM{2 SPACE}
     (J/N) "                                [FAA4]
2260 EIN$=INKEY$ : IF EIN$="" THEN 2260    [885C]
2270 IF UPPER$(EIN$)="J" THEN 860 ELSE L
     OCATE 1,25:PRINT "AUSGABE ALS TAGES
     GANG (J/N)"                            [0908]
2280 EIN$=INKEY$ : IF EIN$="" THEN 2280    [D264]
2290 IF UPPER$(EIN$)="J" THEN 2300 ELSE
     480                                    [2D42]
2300 MODE 2                                 [5AB8]
2310 LOCATE 5,25:PRINT "M O N D : PHASE
     =";:PRINT USING "###.#";MOPHASE;:PR
     INT "%{4 SPACE}AUFGANG: ";:PRINT US
     ING "##.##";MOAUF;:PRINT " UHR{5 SP
     ACE}UNTERGANG: ";:PRINT USING "##.#
     #";MOUNTER;:PRINT " UHR"               [AB4C]
2320 LOCATE 3,1:PRINT "SONNE:{2 SPACE}AU
     FGANG ";:PRINT USING "##.##";SOAUF;
     :PRINT " UHR{3 SPACE}KULMINATION ";
     :PRINT USING "##.##";SOKUL;:PRINT "
      UHR{3 SPACE}UNTERGAMG ";:PRINT USI
     NG "##.##";SOUNTER;:PRINT " UHR"       [6C46]
2330 ORIGIN 20,20,20,620,20,380             [AACE]
2340 CLG 1                                  [DA20]
2350 FOR I=1 TO 5:MOVE 0,I*60:DRAW 600,I
     *60,0:NEXT I                           [D9CE]
2360 MOVE 0,181:DRAW 600,181,0:MOVE 0,17
     9:DRAW 600,179,0                       [369A]
2370 FOR I=1 TO 7:MOVE I*75,0:DRAW I*75,
     360,0:NEXT I                           [26F4]
2380 TAG:MOVE 0,64:PRINT "0";:MOVE 146,6
     4:PRINT "6";:MOVE 292,64:PRINT "12"
     ;:MOVE 442,64:PRINT "18";:MOVE 592,
     64:PRINT "0";:TAGOFF                   [F1FE]
2390 LOCATE 2,13:PRINT "0":LOCATE 1,9:PR
     INT "30":LOCATE 1,5:PRINT "60":LOCA
     TE 1,11:PRINT "+":LOCATE 1,15:PRINT
      "-":LOCATE 1,17:PRINT "30":LOCATE
     1,21:PRINT "60"                        [2BD4]
2400 LOCATE 4,3:PRINT ORT$:LOCATE 30,3:P
     RINT "SONNEN- UND MONDHOEHE":LOCATE
      70,3:PRINT DATUM$                     [A2D4]
2410 MOVE 0,2*SOHOEHE(0)+195:TAG:PRINT "
     SONNE";:TAGOFF:MOVE 0,2*SOHOEHE(0)+
     180                                    [72AA]
2420 FOR I=1 TO 48:DRAW I*12.5,2*SOHOEHE
     (I)+180,0:NEXT I                       [60CA]
2430 MOVE 0,2*MONDHOEHE(0)+195:TAG:PRINT
      "MOND";:TAGOFF:MOVE 0,2*MONDHOEHE(
     0)+180                                 [DB34]
2440 FOR I=1 TO 48:DRAW I*12.5,2*MONDHOE
     HE(I)+180,0:NEXT I                     [88E6]
2450 LOCATE 75,22:PRINT CHR$(237);CHR$(2
     38):LOCATE 75,23:PRINT CHR$(239);CH
     R$(240)                                [398E]
2460 CALL &BB18                             [F676]
2470 LOCATE 1,25:PRINT SPACE$(78):LOCATE
      3,25:PRINT "ANDERES DATUM (J/N)"      [6930]
2480 EIN$=INKEY$ : IF EIN$="" THEN 2480    [986C]
2490 IF UPPER$(EIN$)="J" THEN 860 ELSE L
     OCATE 1,25:PRINT SPACE$(78):LOCATE
     3,25:PRINT "AUSGABE DER DATENTABELL
     E (J/N)"                               [9ECC]
2500 EIN$=INKEY$ : IF EIN$="" THEN 2500    [C550]
2510 IF UPPER$(EIN$)<>"J" THEN 480         [8556]
2520 GOTO 2110                              [C90C]
2530 REM AUSGABE DER MONATLICHEN SONNENT
     ABELLE                                 [683A]
2540 PRINT #2,USING "##";DAY;:PRINT #2,"
     .";:PRINT #2,USING "##";MONTH;:PRIN
     T #2,".";:PRINT #2,USING "##";JAHR;
                                            [D21C]
2550 PRINT #2,USING "####.##";AUFDAEM;:P
     RINT #2,USING "####.##";SOAUF;:PRIN
     T #2,USING "######.##";SOUNTER;:PRI
     NT #2,USING "####.##";UNTDAEM;:PRIN
     T #2,USING "########.##";SOKUL;:PR
     INT #2,USING "######.#";KULHOEHE;     [8818]
2560 PRINT #2,SPACE$(10);:PRINT #2,USING
      "####.#";MOPHASE                      [121E]
2570 DAY=DAY+1                              [9D86]
2580 IF DAY=29 AND MONTH=2 AND (JAHR-4*I
     NT(JAHR/4)=0) THEN 950                 [EFFC]
2590 IF DAY>=29 AND MONTH=2 THEN LOCATE
     2,25:PRINT "NEUER MONAT (J/N)":EIN$
     =INKEY$:IF EIN$="" THEN 2590 ELSE I
     F UPPER$(EIN$)="J" THEN 840 ELSE 26
     30                                     [D5BE]
2600 IF DAY=31 AND (MONTH=4 OR MONTH=6 O
     R MONTH=9 OR MONTH=11) THEN LOCATE
     2,25:PRINT "NEUER MONAT (J/N)":EIN$
     =INKEY$:IF EIN$="" THEN 2600 ELSE I
     F UPPER$(EIN$)="J" THEN 840 ELSE 26
     30                                     [EB34]
2610 IF DAY=32 THEN LOCATE 2,25:PRINT "N
     EUER MONAT (J/N)":EIN$=INKEY$:IF EI
     N$="" THEN 2610 ELSE IF UPPER$(EIN$
     )="J" THEN 840 ELSE 2630               [CF0A]
2620 GOTO 950                               [6CC2]
2630 LOCATE 2,25:PRINT "ANDERER ORT (J/N
     )":EIN$=INKEY$:IF EIN$="" THEN 2630
      ELSE IF UPPER$(EIN$)="J" THEN 790
     ELSE 480                               [F370]
2640 REM AUSGABE DER MONATSTABELLE AUF D
     EM DRUCKER                             [F6E0]
2650 PRINT #8,USING "##";DAY;:PRINT #8,"
     .";:PRINT #8,USING "##";MONTH;:PRIN
     T #8,".";:PRINT #8,USING "##";JAHR;
                                            [D55C]
2660 PRINT #8,USING "####.##";AUFDAEM;:P
     RINT #8,USING "####.##";SOAUF;:PRIN
     T #8,USING "######.##";SOUNTER;:PRI
     NT #8,USING "####.##";UNTDAEM;:PRIN
     T #8,USING "#########.##";SOKUL;:PR
     INT #8,USING "######.#";KULHOEHE;     [0B64]
2670 PRINT #8,SPACE$(4);:PRINT #8,USING
     "####.#";MOPHASE                       [D6E0]
2680 DAY=DAY+1                              [B08A]
2690 IF DAY=29 AND MONTH=2 AND (JAHR-4*I
     NT(JAHR/4)=0) THEN 950                 [F100]
2700 IF DAY>=29 AND MONTH=2 THEN LOCATE
     2,25:PRINT "NEUER MONAT (J/N)":EIN$
     =INKEY$:IF EIN$="" THEN 2700 ELSE I
     F UPPER$(EIN$)="J" THEN 840 ELSE 48
     0                                      [C744]
2710 IF DAY=31 AND (MONTH=4 OR MONTH=6 O
     R MONTH=9 OR MONTH=11) THEN LOCATE
     2,25:PRINT "NEUER MONAT (J/N)":EIN$
     =INKEY$:IF EIN$="" THEN 2710 ELSE I
     F UPPER$(EIN$)="J" THEN 840 ELSE 48
     0                                      [35DE]
2720 IF DAY=32 THEN LOCATE 2,25:PRINT "N
     EUER MONAT (J/N)":EIN$=INKEY$:IF EI
     N$="" THEN 2720 ELSE IF UPPER$(EIN$
     )="J" THEN 840 ELSE 480                [62B4]
2730 GOTO 950                               [71C6]
2740 MODE 0                                 [12C4]
2750 LOCATE 7,12:PRINT "E N D E"            [8366]
2760 CALL &BD18                             [C37C]
2770 GOTO 330                               [6FBE]
2780 MODE 2                                 [2ED0]
2790 LOCATE 20,2:PRINT "ALLGEMEINE ERLAE
     UTERUNGEN":LOCATE 20,3:PRINT STRING
     $(25,"_")                              [8314]
2800 LOCATE 3,6:PRINT "Was bietet das Pr
     ogramm:"                               [9D82]
2810 LOCATE 3,9:PRINT "1. Berechnung des
      Tagesgangs von S O N N E{3 SPACE}u
     nd{3 SPACE}M O N D"                    [8862]
2820 LOCATE 6,11:PRINT "Datenausgabe als
      Tabelle oder Grafik"                  [53DC]
2830 LOCATE 3,14:PRINT "2. Monatliche Ta
     belle fuer die Sonne"                  [07B0]
2840 PRINT STRING$(&50,"_")                 [2282]
2850 PRINT:PRINT " Alle Eingaben muessen
      durch Druecken der <ENTER>-Taste a
```

Listing. Sonne, Mond und Sterne (Fortsetzung)

```
     bgechlossen werden"                       [31F2]
2860 PRINT:PRINT "{12 SPACE}J/N - Abfrag
     en nur durch Druecken von <J> oder
     <N>"                                      [C3EE]
2870 LOCATE 10,25:PRINT "Fortfahren im P
     rogramm durch Druecken irgendeiner
     Taste"                                    [324C]
2880 CALL &BB18                                [3E82]
2890 GOTD 480                                  [85D0]
2900 MODE 2                                    [9CC4]
2910 LOCATE 15,2:PRINT "Erklaerungen zur
      Monatstabelle fuer die Sonne"            [4492]
2920 PRINT STRING$(&50,"_")                    [1980]
2930 PRINT "{11 SPACE}In dieser Darstell
     ung wird fuer alle Orte der Erde":P
     RINT                                      [0132]
2940 PRINT "{11 SPACE}fuer den ausgewaeh
     lten Zeitpunkt angegeben:"                [9E74]
2950 LOCATE 20,9:PRINT "Datum in der For
     m{5 SPACE}TT.MM.JJ"                       [041A]
2960 LOCATE 20,11:PRINT "Daemmerungsbegi
     nn{5 SPACE}hh.mm"                         [A48E]
2970 LOCATE 20,12:PRINT "Sonnenaufgang{9
      SPACE}hh.mm"                             [2850]
2980 LOCATE 20,14:PRINT "Sonnenuntergang
     {7 SPACE}hh.mm"                           [9DBA]
2990 LOCATE 20,15:PRINT "Daemmerungsende
     {7 SPACE}hh.mm"                           [5A6E]
3000 LOCATE 20,17:PRINT "Kulminationszei
     t{6 SPACE}hh.mm"                          [4F5C]
3010 LOCATE 20,18:PRINT "Kulminationshoe
     he in{2 SPACE}**.* Grad"                  [6ECC]
3020 LOCATE 20,20:PRINT "Mondphase in %
     (+ zunehmend{2 SPACE}- abnehmend)"        [B4D8]
3030 PRINT STRING$(&50,"_")                    [B472]
3040 GOTO 2870                                 [DA22]
3050 MODE 2                                    [21BE]
3060 LOCATE 15,2:PRINT "Erklaerungen zur
      Datenausgabe als Tabelle"                [7F62]
3070 PRINT STRING$(&50,"_")                    [2C7A]
3080 PRINT "{2 SPACE}Die Tabelle zeigt f
     uer jeden Ort der Erde im Stundenab
     stand:"                                   [6B40]
3090 LOCATE 20,8:PRINT "Rektaszension ('
     Laenge am Himmel')"                       [3134]
3100 LOCATE 20,9:PRINT "Deklination{3 SP
     ACE}('Breite am Himmel')"                 [8CC8]
3110 LOCATE 20,11:PRINT "Hoehe ueber dem
      Horizont"                                [658E]
3120 LOCATE 20,13:PRINT "Aufgang{3 SPACE
     }um hh.mm Uhr"                            [8E3A]
3130 LOCATE 20,14:PRINT "Untergang um hh
     .mm Uhr"                                  [99A2]
3140 LOCATE 20,16:PRINT "Kulmination der
      Sonne um hh.mm Uhr in **.* Grad Ho
     ehe"                                      [1E38]
3150 LOCATE 20,18:PRINT "Mondphase in %
     (+ zunehmend,{2 SPACE}- abnehmend)"
                                               [6746]
3160 PRINT STRING$(&50,"_")                    [0F7A]
3170 GOTO 2870                                 [752A]
3180 MODE 2                                    [54C6]
3190 LOCATE 15,2:PRINT "Erklaerungen zur
      Datenausgabe als Grafik"                 [3AA0]
3200 PRINT STRING$(&50,"_")                    [E870]
3210 PRINT "{3 SPACE}Die Grafik zeigt de
     n Tagesgang der Sonnen- und Mondhoe
     he fuer jeden Ort"                        [8C18]
3220 LOCATE 10,12:PRINT "Betrachten Sie
     die Grafik. Sie erklaert sich selbs
     t!"                                       [1BBE]
3230 GOTO 2870                                 [6B24]
```

Listing. Sonne, Mond und Sterne (Schluß)

Dem Gärtner zur Freude

Welches Gemüse wo und wie am besten wächst und gedeiht, darüber informiert Sie jetzt Ihr Computer mit unserem Garten-Planungsprogramm.

Nach dem Start des Programms »Mischkultur« erscheint auf dem Bildschirm das Hauptmenü. Geben Sie jetzt eine der zur Auswahl stehenden Kennziffern ein, verzweigt das Programm zum gewählten Unterpunkt. Sofern kein Block-Cursor zu erkennen ist, wird die Eingabe sofort bearbeitet. Anderenfalls drücken Sie ENTER.

Normalerweise beginnt man das Programm mit Punkt 1 (Erstellen des Gartenplans) oder Punkt 4 (Pflanzenübersicht). Die Pflanzenübersicht informiert Sie zunächst über die zur Auswahl stehenden Pflanzen.

Mit der Eingabe eines der hervorgehobenen Buchstaben wählen Sie den entsprechenden Punkt aus.

Möchten Sie Auskunft über eine Einzelpflanze, müssen Sie wenigstens die ersten fünf Buchstaben der Pflanze eingeben. Nun zum eigentlichen Gartenplan: Dazu müssen Sie im Hauptmenü den Punkt 1 anwählen, wenn es sich um eine erste Aufstellung handelt, oder den Punkt 5, wenn Sie einen bereits existierenden Plan weiterbearbeiten wollen.

Jetzt werden Sie aufgefordert, Angaben zur Größe des Gartens, zum Reihenabstand und zur Art der ersten Planzenreihe zu machen. Jede Frage ist mit ENTER zu bestätigen.

Der Computer schlägt vor, welche Pflanze in welcher Reihe am vorteilhaftesten plaziert ist. Sie brauchen nur die Auswahl zu treffen, bis Ihr ganzer Garten bepflanzt ist. Wollen Sie die Bearbeitung abbrechen, so wählen Sie statt einer Reihe »-1« und das Programm verzweigt zurück ins Hauptmenü.

Gefällt Ihnen die »Computerversion« Ihres Gartens und wollen Sie sie speichern, werden Sie aufgefordert, sich einen Namen zu überlegen.

Bei Wahl der Druckerausgabe erhalten Sie Ihren kompletten Gartenplan mit Längenangaben säuberlich auf Papier. Wir wünschen auch in der Praxis gutes Gelingen.

(Hans-Werner Pätz/ja)

```
                 gartplan

Der Garten hat eine Laenge von        5 m.
Der Reihenabstand betraegt           50 cm.
Damit hat der Garten                 10 Reihen.

Reihe  Art    Pflanze

  1     A     Stangenbohnen
  2     C     Neuseelaender Salat
  3     B     Rosenkohl
  4     C     Chicoree
  5     A     Zucchini
  6     C     Spinat
  7     B     Sellerie
  8     C     Kohlrabi
  9     A     Tomaten
 10     C     Kopfsalat/Eissalat
```

Bild. Ein fertig bestücktes Beet

Programm-Steckbrief	
Name:	Gartenplan
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette



```
10   '*********************************     [B910]
20   '*** Der Mischkulturen-Garten ***      [1638]
30   '*********************************     [19D4]
40   '***       copyright          ***      [F840]
50   '***     Werner Paetz         ***      [5BFE]
60   '***       13.2.86            ***      [10D2]
70   '*********************************     [3DDC]
80   '***     5170 Juelich         ***      [D038]
90   '*********************************     [BFE0]
100  '                                      [DEB0]
110 MODE 2                                  [9052]
120 DIM name$(40),gut(40,10),schlecht(40
    ,10),NICHT(40),GUEN(40)                 [BA10]
130 ' Pflanzensortiment                     [103C]
140 DATA 30                                 [FBA4]
150 DATA "ATomaten","AErbsen","AStangenb
    ohnen","AGurken","AZucchini","AKarto
    ffeln"                                  [FC40]
160 DATA "BBuschbohnen","BZwiebeln","BKo
    hl (Rot-,Weiss-,Wirsing)","BRosenkoh
    l","BGruenkohl","BBlumenkohl/Brokkol
    i","BSchwarzwurzel","BRote Beete","B
    Sellerie"                               [1594]
170 DATA "CZwiebeln","CLauch/Porree","CK
    opfsalat/Eissalat","CPflueck-/Schnit
    tsalat","CZuckerhut","CChicoree","CR
    adicchio/roter Zichoriensalat","CEnd
    ivien"                                  [8D46]
180 DATA "CFeldsalat/Rapunzel","CSpinat"
    ,"CNeuseelaender Salat","CMangold","
    CMoehren/Karotten","CKohlrabi","CRet
    tich/Radieschen"                        [0C56]
190 READ anzahl                             [5FE8]
200 FOR i=1 TO anzahl                       [48E2]
210 READ Name$(i)                           [739C]
220 NEXT i                                  [46F8]
230 '                                       [07B8]
240 'Liste des guenstigen Anbaus            [041A]
250 '                                       [E1BC]
260 DATA 8,26,15,18,17,9,8,28,25            [5DD6]
270 DATA 6,4,28,18,9,29,30                  [397E]
280 DATA 6,1,6,23,9,29,18                   [D816]
290 DATA 7,3,7,15,17,14,18,9                [993E]
300 DATA 2,3,8                              [02C4]
310 DATA 4,7,29,25,2                        [911A]
320 DATA 7,1,4,9,29,18,14,15                [BA2E]
330 DATA 5,28,22,18,4,14                    [40A2]
340 DATA 10,1,15,25,18,19,2,7,6,29,30       [4490]
350 DATA 10,1,15,25,18,19,2,7,6,29,30       [F192]
360 DATA 10,1,15,25,18,19,2,7,6,29,30       [E294]
370 DATA 4,17,18,19,29                      [C4FC]
380 DATA 1,15                               [4270]
390 DATA 6,29,19,5,2,7,30                   [F414]
400 DATA 9,7,10,11,13,4,29,17,28,1          [AA66]
410 DATA 5,28,22,18,4,14                    [65A0]
420 DATA 5,28,15,1,18,19                    [84AA]
430 DATA 9,30,29,25,1,3,7,2,4,27            [63AA]
440 DATA 5,1,30,12,14,28                    [FB92]
450 DATA 0                                  [9A46]
460 DATA 4,1,3,18,28                        [C71E]
470 DATA 4,17,9,10,11                       [EA7A]
480 DATA 6,17,9,10,11,28,3                  [246A]
490 DATA 1,8                                [AE18]
500 DATA 8,9,10,11,13,6,1,30,3              [30C8]
510 DATA 1,1                                [BAFC]
520 DATA 6,9,10,11,13,28,30                 [F1B8]
530 DATA 8,16,17,1,2,22,30,27,19            [DCAC]
540 DATA 8,18,19,7,3,15,30,4,2              [05EE]
550 DATA 10,18,7,3,1,9,10,13,28,2,29        [E42C]
560 '                                       [E4C4]
570 'Liste des unguenstigen Anbaus          [B5EC]
580 '                                       [AEC8]
590 DATA 1,3,2,4,6                          [C850]
600 DATA 2,5,3,6,17,1,8                     [CC32]
610 DATA 3,3,2,8,17                         [C5B2]
620 DATA 4,2,1,30                           [BFE0]
630 DATA 6,3,14,15,1                        [0810]
640 DATA 7,3,2,8,17                         [ECC0]
650 DATA 8,6,3,7,2,9,10,11                  [5F5E]
660 DATA 9,2,8,17                           [AC0A]
670 DATA 10,2,8,17                          [6E5C]
680 DATA 11,2,8,17                          [8960]
690 DATA 14,4,17,6,28,25                    [ADBA]
700 DATA 15,1,6                             [772C]
710 DATA 16,6,3,7,2,9,10,11                 [09B6]
720 DATA 17,5,7,3,2,29,14                   [A808]
730 DATA 25,1,14                            [2592]
740 DATA 28,1,14                            [349A]
750 DATA 29,1,17                            [1BA4]
760 DATA 30,1,4                             [3D2E]
770 DATA -1                                 [F4AC]
780 FOR i=1 TO anzahl                       [FAFC]
790 READ b                                  [E0BC]
800 FOR j=1 TO b                            [AFB8]
810 READ gut(i,j)                           [F42A]
820 NEXT j                                  [4306]
830 NEXT i                                  [1B06]
840 flag=0                                  [F986]
850 FOR i=1 TO anzahl                       [FDF8]
860 IF flag=1 AND typ<>i THEN GOTO 910 E
    LSE IF flag=1 THEN 880 ELSE READ typ
                                            [3BB6]
870 IF typ<>i THEN flag=1:GOTO 910          [B2A4]
880 READ menge                              [3D10]
890 FOR j=1 TO menge:READ schlecht(i,j):
    NEXT j                                  [9C4C]
900 flag=0                                  [0E80]
910 NEXT i                                  [4404]
920 WINDOW#1,5,80,6,6                       [4840]
930 WINDOW#2,5,40,11,23                     [82F2]
940 WINDOW#3,45,80,11,23                    [DB66]
950 WINDOW#4,2,80,25,25                     [960A]
960 WINDOW#5,1,80,6,22                      [A7A4]
970 '                                       [73CE]
980 CLS                                     [7946]
990 LOCATE 5,2:PRINT "AUSWAHL FUER GARTE
    NPLAN DES MISCHKULTURENGARTENS"         [78D2]
1000 LOCATE 5,3:PRINT "=================
    ================================"       [E3D0]
1010 LOCATE 8,6:PRINT "1){5 SPACE}ERSTEL
    LEN DES GARTENPLANS"                    [6D04]
1020 LOCATE 8,8:PRINT "2){5 SPACE}AENDER
    N DES GARTENPLANS"                      [76AA]
1030 LOCATE 8,10:PRINT"3){5 SPACE}DRUCKE
    N DES GARTENPLANS"                      [73DE]
1040 LOCATE 8,12:PRINT"4){5 SPACE}PFLANZ
    ENUEBERSICHT"                           [3BB0]
1050 LOCATE 8,14:PRINT"5){5 SPACE}EINLES
    EN EINES GARTENPLANS"                   [5FAC]
1060 LOCATE 8,16:PRINT"0){5 SPACE}E N D
    E"                                      [9014]
1070 LOCATE 8,20:PRINT"{7 SPACE}WELCHE W
    AHL --->{2 SPACE}"                      [CD34]
1080 GOSUB 1380                             [31AA]
1090 IF ein<48 OR ein >53 THEN PRINT CHR
    $(7);CHR$(7):GOTO 980                   [59FA]
1100 ein=ein-48                             [0960]
1110 LOCATE 32,20:PRINT EIN:FOR I=1 TO 2
    00:NEXT I                               [8C80]
1120 ON ein+1 GOTO 3080,1400,2680,2790,1
    130,3410                                [D49A]
1130 CLS                                    [1C8E]
1140 PRINT STRING$(80,"_"):LOCATE 20,3:P
    RINT "A U S W A H L{3 SPACE}D E R{3
    SPACE}P F L A N Z E N"                  [E204]
1150 PRINT STRING$(80,"_")                  [1A2E]
1160 LOCATE 10,8:PRINT "guenstig mit":LO
    CATE 55,8:PRINT "unguenstig mit"        [F77C]
1170 PRINT STRING$(80,"_")                  [4032]
1180 MOVE 0,30:DRAW 640,30:MOVE 330,30:D
    RAW 330,284                             [0B9E]
1190 PRINT#4,"A) -Reihe{5 SPACE}B) -Reih
    e{5 SPACE}C) -Reihe{5 SPACE}E) inze
    lpflanze{5 SPACE}M) enue"               [9E22]
1200 GOSUB 1380                             [539E]
1210 IF ein=77 THEN 980                     [9A14]
1220 IF ein=65 THEN anfang=1:auswahl=6:G
    OTO 1270                                [2D7E]
1230 IF ein=66 THEN anfang=7:auswahl=15:
    GOTO 1270                               [CBEE]
1240 IF ein=67 THEN anfang=16:auswahl=30
    :GOTO 1270                              [D64C]
1250 IF ein=69 THEN CLS#4:INPUT#4,"Welch
    e Pflanze :";Pflanze$:FOR Ij=1 TO a
    nzahl:IF UPPER$(LEFT$(pflanze$,5))=
    UPPER$(LEFT$(MID$(name$(ij),2),5))
    THEN anfang=ij:auswahl=ij:GOTO 1270
     ELSE NEXT ij:CLS#4:GOTO 1250           [A81E]
1260 GOTO 1340                              [6414]
1270 FOR i=anfang TO auswahl                [C2B4]
1280 CLS#1:PRINT#1, LEFT$(Name$(I),1);"
    - Reihe",MID$(name$(i),2)               [2EEA]
1290 CLS#2:CLS#3                            [842A]
1300 FOR k=1 TO 10:IF gut(i,k)=0 THEN 13
    10 ELSE PRINT#2,MID$(name$(gut(i,k)
    ),2):NEXT k                             [5724]
1310 FOR j=1 TO 10:IF schlecht(i,j)=0 TH
    EN 1315 ELSE PRINT#3,"{3 SPACE}";MI
    D$(name$(schlecht(i,j)),2):NEXT j       [BDE0]
1315 IF anfang=auswahl THEN CLS#4:PRINT#
    4,"{6 SPACE}M) enue{9 SPACE}E) inze
    lpflanze ":GOSUB 1360: IF ein=69 TH
    EN 1250 ELSE IF ein=77 THEN 980 ELS
    E GOTO 1340                             [8468]
1320 CLS#4:PRINT#4,"{6 SPACE}M) enue{9 S
    PACE}W) eiter ":GOSUB 1360: IF ein=
    77 THEN 980                             [0B9A]
1330 NEXT i                                 [8E5E]
1340 CLS#4:PRINT CHR$(7);CHR$(7):GOTO 11
    90                                      [1666]
1350 REM                                    [3D9A]
1360 REM Eingabe-Routine                    [4A58]
1370 REM                                    [2F9E]
```

Listing. Optimale Nutzung Ihrer Gemüsebeete

```
1380 ein$=INKEY$:IF ein$="" THEN 1380 EL
     SE ein=ASC(UPPER$(ein$)):RETURN        [0B6E]
1390 REM                                    [01A2]
1400 CLS                                    [258E]
1410 PRINT STRING$(80,"_"):LOCATE 12,3:P
     RINT"E R S T E L L E N{3 SPACE}E I
     N E S{3 SPACE}G A R T E N P L A N S
     "                                      [F7AC]
1420 PRINT STRING$(80,"_")                  [1F2E]
1430 IF aflag=1 THEN 2510                   [1B6E]
1440 LOCATE 5,6:INPUT "Wie lang ist die
     zu bepflanzende Flaeche (in m){9 SP
     ACE}";laenge                           [A660.
1450 LOCATE 5,8:INPUT "Wie gross soll de
     r Reihenabstand sein (40,50 oder 60
      cm)";abst:IF ABST<40 OR ABST>60 TH
     EN 1450 ELSE abst=abst/100             [5070]
1460 LOCATE 5,10:INPUT "Welches ist die
     erste Reihe (A, B oder C){15 SPACE}
     ";reih$:reih$=UPPER$(reih$)            [5BE8]
1470 IF REIH$="A" OR REIH$="B" OR REIH$=
     "C" THEN 1480 ELSE 1460                [17F4]
1480 reihanz=CINT(laenge/abst)              [5ABE]
1490 IF reihanz<33 THEN 1530 ELSE LOCATE
      2,20: PRINT "Bei einem Reihenabsta
     nd von";CINT(100*abst);"cm darf das
      Gartenstueck nur";32*abst;"m lang
     sein."                                 [31C8]
1500 LOCATE 15,22:PRINT "Teilen Sie die
     Gartenflaeche in mehrere Stuecke"      [BD1E]
1510 LOCATE 14,24:PRINT "**** Zum Fortfa
     hren beliebige Taste druecken ****"
                                            [094C]
1520 CALL &BB18:GOTO 1400                   [601E]
1530 IF (reih$="A") THEN aanz=INT((reiha
     nz+3)/4):canz=INT(reihanz/2):banz=r
     eihanz-aanz-canz:GOTO 1580             [97C2]
1540 IF (reih$="B") THEN banz=INT((reiha
     nz+3)/4):canz=INT(reihanz/2):aanz=r
     eihanz-banz-canz:GOTO 1580             [61C8]
1550 aanz=INT((reihanz+2)/4)                [6E20]
1560 banz=INT(reihanz/4)                    [86C8]
1570 canz=INT((reihanz+1)/2)                [A622]
1580 LOCATE 2,15:PRINT"Der Garten wird i
     nsgesamt ";reihanz;"Reihen haben ("
     ;aanz;"x A, ";banz;"x B und ";canz;
     "x C)"                                 [216E]
1590 DIM reihe$(reihanz),reihe(reihanz)     [C5AE]
1600 LOCATE 5,24:PRINT SPACE$(70):LOCATE
      5,24:PRINT "Bepflanzen der Reihe 1
      ..";reihanz;"{4 SPACE}Ende=-1";       [37E0]
1610 IF aflag=0 THEN LOCATE 50,24:PRINT
     "von vorne an = 0";                    [5BB8]
1620 INPUT reihanf                          [63EC]
1630 IF reihanf<0 THEN 980                  [1CE4]
1640 IF REIHANF>REIHANZ THEN PRINT CHR$(
     7):GOTO 1600                           [6ECA]
1650 aflag=1                                [33AA]
1660 IF reihanf=0 THEN 2560                 [D344]
1670 i=reihanf                              [00E2]
1680 IF reih$="C" THEN 1710                 [60CC]
1690 IF reih$="A" THEN 1720                 [77CC]
1700 IF reih$="B" THEN 1730                 [1AC0]
1710 IF (i+1) MOD 2=0 THEN buch$="C":GOT
     O 1740 ELSE IF (i+2) MOD 4=0 THEN b
     uch$="A":GOTO 1740 ELSE buch$="B":G
     OTO 1740                               [0FB8]
1720 IF (i+3) MOD 4=0 THEN buch$="A":GOT
     O 1740 ELSE IF i MOD 2=0 THEN buch$
     ="C":GOTO 1740 ELSE buch$="B":GOTO
     1740                                   [9C62]
1730 IF (i+3) MOD 4=0 THEN buch$="B":GOT
     O 1740 ELSE IF i MOD 2=0 THEN buch$
     ="C":GOTO 1740 ELSE buch$="A"          [E4A6]
1740 REM                                    [32A0]
1750 FOR NI=1 TO 40: NICHT(NI)=0: GUEN(N
     I)=0:  NEXT NI                         [DE66]
1760 IF buch$="A" THEN 1790                 [5FCA]
1770 IF buch$="B" THEN 2010                 [FDB2]
1780 IF buch$="C" THEN 2230                 [04BE]
1790 FOR RV=-4 TO 4                         [4CAA]
1800 IF RV=0 THEN 1890                      [F7DC]
1810 IF (REIHANF+RV)<0 OR (REIHANF+RV)>R
     EIHANZ THEN 1890                       [6770]
1820 IF REIHE(REIHANF+RV)=0 THEN 1890       [07AC]
1830 FOR RVA=1 TO 6                         [F4C6]
1840 FOR RVJ=1 TO 10                        [1930]
1850 IF SCHLECHT(RVA,RVJ) = REIHE(REIHAN
     F+RV) THEN NICHT(RVA)=1                [05B6]
1860 IF GUT(RVA,RVJ) = REIHE(REIHANF+RV)
      THEN GUEN(RVA)=1                      [B06E]
1870 NEXT RVJ                               [1182]
1880 NEXT RVA                               [C772]
1890 NEXT RV                                [21F2]
1900 IF REIH0=1 THEN LOCATE 5,24:PRINT C
     HR$(7);CHR$(7);"*** ZUM FORTFAHREN
     IRGENDEINE TASTE DRUECKEN ***" ELSE
      GOTO 1920                             [B802]
1910 CALL &BB18:LOCATE 5,24:PRINT SPACE$
     (60)                                   [1982]
1920 CLS#5:LOCATE#5, 2,1:PRINT#5, "Die R
     eihe ";reihanf;" ist eine ";buch$;"
      - Reihe, damit stehen folgende Pfl
     anzen zur Auswahl:"                    [AEDE]
1930 FOR IND=1 TO 6                         [C9AC]
1940 IF NICHT(IND)=1 THEN 1990              [57DE]
1950 LOCATE#5,10,IND+2                      [77E0]
1960 IF GUEN(IND)=1 THEN PRINT#5,CHR$(24
     );                                     [BDB6]
1970 PRINT#5,IND;" :{2 SPACE}";MID$(NAME
     $(IND),2)                              [AC9E]
1980 IF GUEN(IND)=1 THEN PRINT#5,CHR$(24
     );                                     [F3BA]
1990 NEXT IND                               [945A]
2000 GOTO 2440                              [3A0A]
2010 FOR RV=-4 TO 4                         [458E]
2020 IF RV=0 THEN 2110                      [BDB6]
2030 IF (REIHANF+RV)<0 OR (REIHANF+RV)>R
     EIHANZ THEN 2110                       [994A]
2040 IF REIHE(REIHANF+RV)=0 THEN 2110       [5A86]
2050 FOR RVB=7 TO 15                        [CF2A]
2060 FOR RVJ=1 TO 10                        [D626]
2070 IF SCHLECHT(RVB,RVJ) = REIHE(REIHAN
     F+RV) THEN NICHT(RVB)=1                [2BB0]
2080 IF GUT(RVB,RVJ) = REIHE(REIHANF+RV)
      THEN GUEN(RVB)=1                      [9368]
2090 NEXT RVJ                               [C378]
2100 NEXT RVB                               [D458]
2110 NEXT RV                                [4CD6]
2120 IF REIH0=1 THEN LOCATE 5,24:PRINT C
     HR$(7);CHR$(7);"*** ZUM FORTFAHREN
     IRGENDEINE TASTE DRUECKEN ***" ELSE
      GOTO 2140                             [1DEE]
2130 CALL &BB18:LOCATE 5,24:PRINT SPACE$
     (60)                                   [BC78]
2140 CLS#5:LOCATE#5, 2,1:PRINT#5, "Die R
     eihe ";reihanf;" ist eine ";buch$;"
      - Reihe, damit stehen folgende Pfl
     anzen zur Auswahl:"                    [1ED4]
2150 FOR IND=7 TO 15                        [C30E]
2160 IF NICHT(IND)=1 THEN 2210              [01B8]
2170 LOCATE#5,10,(IND-6)+2                  [113E]
2180 IF GUEN(IND)=1 THEN PRINT#5,CHR$(24
     );                                     [92AC]
2190 PRINT#5,IND-6;" :{2 SPACE}";MID$(NA
     ME$(IND),2)                            [185A]
2200 IF GUEN(IND)=1 THEN PRINT#5,CHR$(24
     );                                     [069E]
2210 NEXT IND                               [A53E]
2220 GOTO 2440                              [3C12]
2230 FOR RV=-4 TO 4                         [4796]
2240 IF RV=0 THEN 2330                      [C5C6]
2250 IF (REIHANF+RV)<0 OR (REIHANF+RV)>R
     EIHANZ THEN 2330                       [915A]
2260 IF REIHE(REIHANF+RV)=0 THEN 2330       [4E96]
2270 FOR RVC=16 TO 30                       [008E]
2280 FOR RVJ=1 TO 10                        [FC2E]
2290 IF SCHLECHT(RVC,RVJ) = REIHE(REIHAN
     F+RV) THEN NICHT(RVC)=1                [EBBC]
2300 IF GUT(RVC,RVJ) = REIHE(REIHANF+RV)
      THEN GUEN(RVC)=1                      [2762]
2310 NEXT RVJ                               [926E]
2320 NEXT RVC                               [E462]
2330 NEXT RV                                [5ADE]
2340 IF REIH0=1 THEN LOCATE 5,24:PRINT C
     HR$(7);CHR$(7);"*** ZUM FORTFAHREN
     IRGENDEINE TASTE DRUECKEN ***" ELSE
      GOTO 2360                             [EDFE]
2350 CALL &BB18:LOCATE 5,24:PRINT SPACE$
     (60)                                   [9680]
2360 CLS#5:LOCATE#5, 2,1:PRINT#5, "Die R
     eihe ";reihanf;" ist eine ";buch$;"
      - Reihe, damit stehen folgende Pfl
     anzen zur Auswahl:"                    [28DC]
2370 FOR IND=16 TO 30                       [4370]
2380 IF NICHT(IND)=1 THEN 2430              [21C8]
2390 LOCATE#5,10,(IND-15)+2                 [05A6]
2400 IF GUEN(IND)=1 THEN PRINT#5,CHR$(24
     );                                     [EEA2]
2410 PRINT#5,IND-15;" :{2 SPACE}";MID$(N
     AME$(IND),2)                           [39B0]
2420 IF GUEN(IND)=1 THEN PRINT#5,CHR$(24
     );                                     [34A6]
2430 NEXT IND                               [BF46]
2440 LOCATE 5,24:PRINT SPACE$(70):LOCATE
      5,24:INPUT"Welche Pflanze (bitte K
     ennziffer [-1 = ENDE]) ---> ";ein      [D7DA]
2450 IF ein=0 THEN z=0: GOTO 2500           [B026]
2460 IF ein = -1 THEN reih0=0:GOTO 980      [DC34]
2470 IF buch$="A" THEN z=ein:GOTO 2500      [16C0]
2480 IF buch$="B" THEN z=ein+6:GOTO 2500
                                            [8286]
```

Listing. Optimale Nutzung Ihrer Gemüsebeete (Fortsetzung)



```
2490 IF buch$="C" THEN z=ein+15                    [A236]
2500 reihe$(reihanf)=MID$(name$(z),2):re
     ihe(reihanf)=z                                 [05D8]
2510 CLS#5:FOR i=1 TO reihanz:IF i<17 TH
     EN LOCATE 2,5+i:PRINT USING "##";i;
     :PRINT" :{2 SPACE}";LEFT$(reihe$(i)
     ,30) ELSE LOCATE 42,5+i-16:PRINT US
     ING "##";i;:PRINT" :{2 SPACE}";LEFT
     $(reihe$(i),30)                                [9A04]
2520 NEXT i                                         [9062]
2530 IF flaga=1 THEN RETURN                         [F8A2]
2540 FOR ENDE=1 TO REIHANZ:IF REIHE(ENDE
     )=0 THEN 2550 ELSE NEXT ENDE:GOTO 2
     600                                            [7634]
2550 IF REIH0=1 THEN 2580 ELSE 1600                 [039E]
2560 REM GARTENPLAN MIT REIHE 1 BEGINNEN
     !!                                             [FFDA]
2570 REIH0=1                                        [BD68]
2580 REIHANF=REIHANF+1                              [AB04]
2590 GOTO 1670                                      [6F2E]
2600 LOCATE 5,24:PRINT "**** A L L E{3 S
     PACE}R E I H E N{3 SPACE}S I N D{3
     SPACE}B E P F L A N Z T ****"                  [DEDE]
2610 LOCATE 18,25:PRINT "D) RUCKEN{6 SPA
     CE}A) ENDERN{7 SPACE}M) ENUE"                  [F6CC]
2620 EIN$=INKEY$:IF EIN$="" THEN 2620 EL
     SE EIN$=UPPER$(EIN$)                           [512A]
2630 IF EIN$="M" THEN 980                           [40F0]
2640 IF EIN$="A" THEN 2680                          [1C38]
2650 IF EIN$="D" THEN 2790                          [2F44]
2660 LOCATE 18,25:PRINT "{2 SPACE}F A L
     S C H E{3 SPACE}E I N G A B E{7 SPA
     CE}":FOR I=1 TO 500:NEXT:GOTO 2610             [3F26]
2670 REM                                            [32A6]
2680 REM Aendern des Gartenplans                    [DC58]
2690 REM                                            [04AA]
2700 CLS                                            [1896]
2710 PRINT STRING$(80,"_"):LOCATE 12,3:P
     RINT"A E N D E R N{3 SPACE}E I N E
     S{3 SPACE}G A R T E N P L A N S"               [1FD2]
2720 PRINT STRING$(80,"_")                          [4636]
2730 IF aflag=0 THEN CLS:LOCATE 5,12:PRI
     NT "Es ist noch keine Reihe bepflan
     zt":FOR i=1 TO 2000:NEXT i:GOTO 980
                                                    [B7AE]
2740 flaga=1:GOSUB 2510:flaga=0                     [7634]
2750 LOCATE 5,24:INPUT"A e n d e r n{4 S
     PACE}w e l c h e r{3 SPACE}R e i h
     e{2 SPACE}oder -1 fuer Ende: ";reih
     anf                                            [D9AE]
2760 IF reihanf=0 THEN PRINT CHR$(7);CHR
     $(7):GOTO 2750                                 [AA64]
2770 IF reihanf<0 THEN 980                          [D3F0]
2780 flaga=1:GOSUB 1670:flaga=0:GOTO 275
     0                                              [1E0A]
2790 REM                                            [10AC]
2800 REM Druck-Routine                              [4DF4]
2810 REM                                            [549E]
2820 CLS                                            [CA9C]
2830 PRINT STRING$(80,"_"):LOCATE 20,3:P
     RINT "D R U C K E N{3 SPACE}D E S{3
      SPACE}G A R T E N P L A N S"                  [9884]
2840 PRINT STRING$(80,"_")                          [763C]
2850 IF PLAN$="" THEN PLAN$="GARTENPLAN"
                                                    [C632]
2860 LOCATE 5,24:PRINT "B I T T E{3 SPAC
     E}D R U C K E R{4 SPACE}E I N S C H
      A L T E N !!"                                 [1D64]
2870 LOCATE 10,25:PRINT "danach beliebig
     e Taste druecken":CALL &BB18                   [8E34]
2880 PRINT#8,CHR$(11)                               [9774]
2890 PRINT#8,CHR$(27);CHR$(49)                      [B078]
2900 PRINT#8,STRING$(80,"_"):PRINT#8:PRI
     NT#8,SPACE$(30);plan$:PRINT#8,STRIN
     G$(80,"_")                                     [B922]
2910 PRINT#8:PRINT#8,SPACE$(10);"Der Gar
     ten hat eine Laenge von{5 SPACE}";l
     aenge;"m."                                     [74BA]
2920 PRINT#8,SPACE$(10);"Der Reihenabsta
     nd betraegt{8 SPACE}";INT(abst*100)
     ;"cm."                                         [F6F8]
2930 PRINT#8,SPACE$(10);"Damit hat der G
     arten{15 SPACE}";:PRINT#8,USING"##"
     ;reihanz;:PRINT#8," Reihen."                   [120C]
2940 PRINT#8:PRINT#8:PRINT#8,SPACE$(10);
     "Reihe{2 SPACE}Art{4 SPACE}Pflanze"
                                                    [5E58]
2950 PRINT#8,SPACE$(10);STRING$(70,"_")             [FE06]
2960 FOR i=1 TO reihanz                             [5146]
2970 PRINT#8:PRINT#8,SPACE$(10);:PRINT#8
     ,USING"###";i;:PRINT#8,"{5 SPACE}";
     LEFT$(name$(reihe(i)),1);"{5 SPACE}
     ";reihe$(i)                                    [BE22]
2980 NEXT i                                         [B476]
2990 PRINT STRING$(80,"_"):LOCATE 30,3:P
     RINT plan$:PRINT STRING$(80,"_")               [D630]
3000 PRINT:PRINT:PRINT:PRINT"Der Garten
     hat eine Laenge von";laenge;"m."               [7CC6]
3010 PRINT:PRINT"Der Reihenabstand betra
     egt";INT(abst*100);"cm."                       [786C]
3020 PRINT:PRINT"Damit hat der Garten{6
     SPACE}";reihanz;"Reihen."                      [A432]
3030 PRINT:PRINT:PRINT:PRINT"Reihe{8 SPA
     CE}Art{10 SPACE}Pflanze"                       [FD60]
3040 FOR i=1 TO reihanz                             [2632]
3050 PRINT:PRINT USING"###";i;:PRINT"{11
      SPACE}";LEFT$(name$(reihe(i)),1);"
     {11 SPACE}";reihe$(i)                          [1AE6]
3060 NEXT i                                         [8962]
3070 PRINT "zum Fortfahren eine Taste dr
     uecken":CALL &BB18                             [3B84]
3080 REM                                            [4A9E]
3090 REM ENDE DES PROGRAMMS                         [0BC0]
3100 REM                                            [3090]
3110 CLS                                            [208E]
3120 PRINT STRING$(80,"_"):LOCATE 20,3:P
     RINT "B E E N D E N{3 SPACE}D E S{3
      SPACE}P R O G R A M M S"                      [48B2]
3130 PRINT STRING$(80,"_")                          [322E]
3140 LOCATE 20,10:PRINT "1 .... Abspeich
     ern des Gartenplans"                           [4CC4]
3150 LOCATE 20,12:PRINT "2 .... Zurueck
     zum Hauptmenue"                                [B350]
3160 LOCATE 20,14:PRINT "0 .... E N D E"
                                                    [DF08]
3170 LOCATE 27,20:PRINT "Kennziffer --->
     {6 SPACE}"                                     [FB38]
3180 GOSUB 1380                                     [3CB0]
3190 ein=ein-48:IF ein<0 OR ein >2 THEN
     PRINT CHR$(7);CHR$(7):GOTO 3170                [EC80]
3200 LOCATE 45,20:PRINT ein                         [90AE]
3210 IF ein = 2 THEN 980                            [A120]
3220 IF ein = 0 THEN CLS:LOCATE 20,12:PR
     INT "E N D E":LOCATE 1,23:END                  [E7BE]
3230 REM Abspeichern                                [EBA0]
3240 LOCATE 10,22:INPUT "Welcher Name :"
     ;plan$                                         [695E]
3250 LOCATE 10,22:PRINT "Bitte das Band
     an die richtige Stelle spulen"                 [D466]
3260 LOCATE 10,23:PRINT "danach 'REC/PLA
     Y' und irgendeine Taste druecken":C
     ALL &BB18                                      [B150]
3270 OPENOUT "!gartplan"                            [D7E8]
3280 PRINT#9,PLAN$                                  [18A2]
3290 PRINT#9,laenge                                 [F4DE]
3300 PRINT#9,abst                                   [514A]
3310 PRINT#9,reihanz                                [2CDA]
3320 PRINT#9,reih$                                  [4A92]
3330 FOR i=1 TO reihanz                             [2336]
3340 PRINT#9,reihe$(i)                              [35D4]
3350 PRINT#9,reihe(i)                               [608E]
3360 NEXT i                                         [CC68]
3370 CLOSEOUT                                       [42B6]
3380 LOCATE 10,23:PRINT SPACE$(60)                  [AFCC]
3390 LOCATE 10,22:PRINT "{4 SPACE}Abspei
     chern{2 SPACE}des Gartenplans beend
     et{6 SPACE}"                                   [4032]
3400 GOTO 3130                                      [9D0E]
3410 REM                                            [4F98]
3420 REM EINLESEN EINES GARTENPLANS                 [1D66]
3430 REM                                            [4D9C]
3440 CLS:LOCATE 10,12:PRINT "Bitte das B
     and an die richtige Stelle spulen"             [E39E]
3450 LOCATE 10,14:PRINT "{3 SPACE}danach
      'PLAY' und irgendeine Taste drueck
     en":CALL &BB18                                 [C000]
3460 OPENIN "!gartplan"                             [6E28]
3470 INPUT#9,PLAN$                                  [EAAA]
3480 INPUT#9,laenge                                 [2CE6]
3490 INPUT#9,abst                                   [9664]
3500 INPUT#9,reihanz                                [A0E2]
3510 INPUT#9,reih$                                  [CC9A]
3520 DIM reihe$(reihanz),reihe(reihanz)             [ADA4]
3530 FOR i=1 TO reihanz                             [413A]
3540 INPUT#9,reihe$(i)                              [D6DE]
3550 INPUT#9,reihe(i)                               [7298]
3560 NEXT I                                         [6A2C]
3570 CLOSEIN                                        [50F8]
3580 aflag=1                                        [7EB2]
3590 GOTO 980                                       [9ED6]
```

Listing. Optimale Nutzung Ihrer Gemüsebeete (Schluß)



Volle Übersicht

Endlose Zahlenkolonnen sind meistens sehr schwer lesbar. Dieses Programm hilft Ihnen, statistische Daten grafisch übersichtlich darzustellen.

Egal, wie auch immer Sie Ihre Daten veranschaulichen wollen, »Statistik« hält vier verschiedene Arten für Sie bereit: Kuchen-Diagramm, Balken-Diagramm, Trend-Darstellung und Break-even-Point. Zugunsten der höheren Auflösung und großzügigerer Beschriftung erfolgt die Bildschirm-Ausgabe im Modus 2. Durch die verwendeten Farbkombinationen eignet sich das Programm für den Einsatz sowohl auf einem Farb- als auch auf einem Grün-Monitor. Anwendungen für das Programm finden sich nicht nur im privaten Bereich: auch Kleinbetrieben und Selbständigen kann es ein wertvolles Hilfsmittel für die Planung sein, da es automatisch aus den Daten vergangener Zeiträume Trends für die nähere Zukunft berechnet. Stellt man zwei verschiedene Daten einander gegenüber, lassen sich beispielsweise Kosten und Umsätze miteinander vergleichen. Beispiele für die Bildschirm-Darstellung von Balken- und Trend-Diagramm finden Sie in Bild 1 und 2. Dabei wird auch noch zwischen mehreren Produkten und Kostenarten unterschieden.

(Klaus Eschemann/ja)

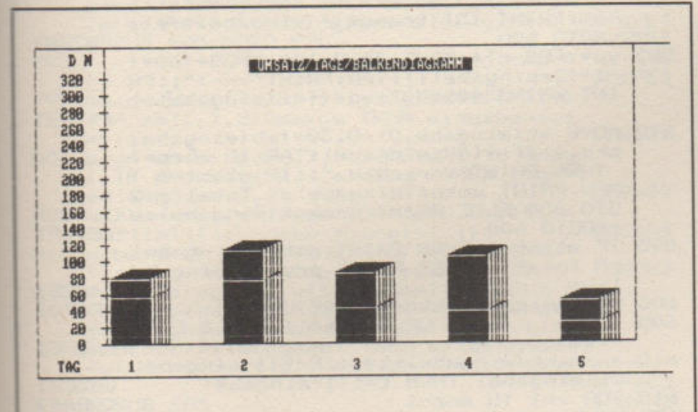


Bild 1. Umsätze zweier Artikel als Balkendiagramm

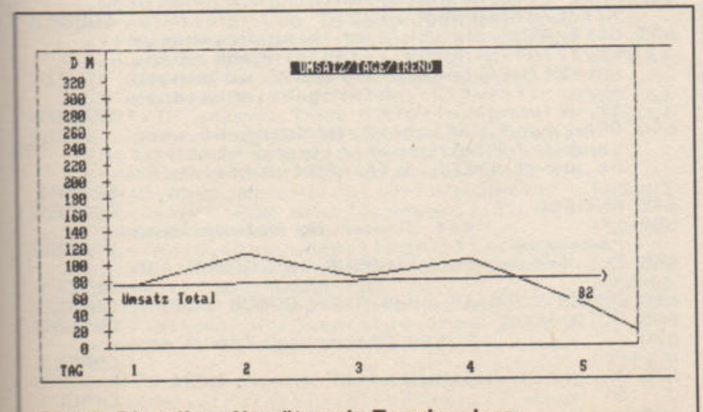


Bild 2. Dieselben Umsätze als Trendanalyse

Programm-Steckbrief	
Name:	Statistik
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
100 GOTO 915                                  [B552]
105 '                  *** UP Beschriften
     ***                                      [48D6]
110 ueber$=umko$(eingabe)+"/"+zeige$+"/"
    +hm$(hm)+umr$(eingabe):ueber$=UPPER$
    (ueber$)                                  [5160]
115 xanf=80-anz(eingabe):schritt=INT((64
    0-xanf)/anz(eingabe)):tf=25-(anz(ein
    gabe)/2):wt=((22-anz(eingabe))*2)+an
    z(eingabe):xend=(xanf*2)+((anz(einga
    be)-1)*schritt)                           [308A]
120 a=INT((xanf+((anz(eingabe))*schritt)
    -schritt)/16)+INT((xanf)/8):ON pm GO
    SUB 2140,2150,2140,2155,2140,2160,21
    40,2165                                   [1510]
125 MODE 2:LOCATE a-INT(LEN(ueber$)/2),2
    :PRINT"{CTRL X} ";ueber$;" {CTRL X}"
    :PLOT 0,0:DRAWR 0,399:DRAWR 639,0:DR
    AWR 0,-399:DRAWR-639,0                    [BFC4]
130 FOR a=0 TO anz(eingabe)-1:READ zeit$
    :MOVE(wt/2)+xanf+(schritt*a)-(LEN(ze
    it$)*4),20:TAG:PRINT zeit$;:NEXT:MOV
    E xanf-40,390:TAG:PRINT"D M";:MOVE x
    anf-5,400:DRAW xanf-5,40,3                [2B10]
135 FOR a=360 TO 40 STEP -20 :uk$=STR$(I
    NT((a-40)*f))                             [6DC2]
140   MOVE xanf-10,a:DRAWR 8,0:MOVE 2,a+
    6:TAG:IF LEN(uk$)>4 THEN PRINT RIGHT
    $(bl$+MID$(uk$,1,LEN(uk$)-3),erl1(ei
    ngabe)-3);".";RIGHT$(uk$,3); ELSE PR
    INT RIGHT$(bl$+uk$,erl1(eingabe)+((L
    EN(maxdm$(eingabe))-1)\4));               [48C4]
145 NEXT:MOVE 2,46:TAG:PRINT SPC((erl1(e
    ingabe)-1)+((LEN(maxdm$(eingabe))-1)
    \4));"0";:MOVE xanf,40:DRAW xend,40:
    DRAWR 0,360                               [2320]
150 MOVE 20,20:TAG:PRINT UPPER$(zeig$(pm
    ));:RETURN                                [A1A2]
155 '              *** UP Vorschau (parabol.
     Trend) ***                               [7116]
160 ss=0:xss=0:x2ss=0:x2su=0:x3s=0:x4s=0
    ::s=0:xs=0:x2s=0:x2=0:x3=0:x4=0           [8854]
165 anz=anz(eingabe):t=t(eingabe)             [FC44]
170 IF anz MOD 2 THEN ordnz=1:ordan=((an
    z-1)/2)*-1 ELSE ordnz=2:ordan=(anz-1
    )*-1                                      [AFB2]
175 x=ordan:FOR a=1 TO anz:s=hwert(einga
    be,start,a-1):xs=x*s:x2s=(x^2)*s:x2=
    x^2:x3=x^3:x4=x^4:ss=ss+s:xss=xss+xs
    :x2ss=x2ss+x2s:x2su=x2su+x2:x3s=x3s+
    x3:x4s=x4s+x4:x=x+ordnz:NEXT              [20FE]
180 b=xss/x2su:c1=(x2ss-((x2su/anz)*ss))
    :c2=(x4s-(x2su/anz)*x2su):IF c2=0 TH
    EN c2=1                                   [C0D2]
185 diff=0:diff=hwert(eingabe,start,anz-
    1)-hwert(eingabe,start,0)                 [9CE0]
190 c=c1/c2:a=(ss-(x2su*c))/anz:x=ordan:
    FOR y=0 TO anz-1:hwert(eingabe,7,y)=
    a+(b*x)+(c*(x^2))+diff                    [F0AC]
195 mh(eingabe,7,y)=(hwert(eingabe,7,y)/
    f(eingabe)/t)*INT((SGN(hwert(eingabe
    ,7,y))+1)/2)                              [ABC6]
200 IF mh(eingabe,7,y)<=0 THEN mh(eingab
    e,7,y)=2                                  [5FB4]
205 x=x+ordnz:NEXT                            [1D6A]
210 RETURN                                    [CD26]
215 '              *** UP Maximal darzustelle
    nder Wert ***                             [9F06]
220 umr$(eingabe)="":t=1:IF anz(eingabe)
    >15 THEN erl1=4 ELSE erl1=5               [51E0]
225 FOR a=start TO proko-1:FOR b=0 TO an
    z(eingabe)-1:maxdm(eingabe,b)=maxdm(
    eingabe,b)+hwert(eingabe,a,b):NEXT:N
    EXT                                       [4E4A]
230 FOR x=0 TO anz(eingabe)-2                 [B890]
235 FOR y=x+1 TO anz(eingabe)-1               [81E2]
240 IF maxdm(eingabe,x)<=maxdm(eingabe,y
    ) THEN hi=maxdm(eingabe,x):maxdm(ein
    gabe,x)=maxdm(eingabe,y):maxdm(einga
    be,y)=hi                                  [884E]
245 NEXT:NEXT:maxdm=maxdm(eingabe,0):max
    dm$=STR$(maxdm)                           [FF8E]
250 FOR a=1 TO LEN(maxdm$):IF MID$(maxdm
    $,a,1)="+" THEN li=a:le=VAL(RIGHT$(m
    axdm$,2)) ELSE IF MID$(maxdm$,a,1)="
    ." THEN pui=a                             [8FC8]
```

Listing. Umsatz, Kosten, Trends und Analysen

```
255 NEXT a:IF pui=0 THEN pui=li-2                 [1F9C]
260 IF le THEN dm=VAL(MID$(maxdm$,pui-1,
    pui))*10^(li-2-pui)+VAL(MID$(maxdm$,
    pui+1,LEN(maxdm$)-li+1)):maxdm$="":m
    axdm$=STR$(dm):maxdm$=maxdm$+STRING$
    (le-(li-pui-2),"0")                          [1396]
265 IF LEN(maxdm$)-1>erll THEN t=((LEN(m
    axdm$)-1)-erll)/3 ELSE GOTO 295              [009A]
270 IF INT(t)<t THEN t=INT(t)+1                  [79E8]
275 ON t GOSUB 280,285,290:tl=t*3:t=10^t
    l:maxdm=VAL(LEFT$(maxdm$,LEN(maxdm$)
    -(tl))):GOTO 295                             [73A4]
280 umr$(eingabe)=" IN TSD DM":RETURN            [B8B8]
285 umr$(eingabe)=" IN MIO DM":RETURN            [F0B6]
290 umr$(eingabe)=" IN MRD DM":RETURN            [B2AA]
295 t(eingabe)=t:erll(eingabe)=erll+1:RE
    TURN                                         [C1D0]
300 '             *** UP Berechnen Darstell
    ungswerte ***                                [77C8]
305 DEG:tg=TAN(62):sw=SIN(30):cw=COS(30)
    :off=40:f=maxeing(eingabe)/320:f(ein
    gabe)=f                                      [43F4]
310 xanf=80-anz(eingabe):schritt=INT((64
    0-xanf)/anz(eingabe))                        [0D4A]
315 FOR x=0 TO anz(eingabe)-1:mh(eingabe
    ,6,x)=hwert(eingabe,6,x)/f(eingabe)/
    t(eingabe):st(eingabe,x)=x*schritt+x
    anf:NEXT                                     [260E]
320 FOR sle=0 TO proko(eingabe)-1                [A7EA]
325 FOR x=0 TO anz(eingabe)-1                    [0398]
330 mh=hwert(eingabe,sle,x)/f(eingabe)/t
    (eingabe):mh(eingabe,sle,x)=ROUND(mh
    ):NEXT:NEXT:RETURN                           [5190]
335 '                    *** UP Maskenaufbau
     ***                                         [F5EC]
340 MODE 2:BORDER 9:LOCATE 15,2:PRINT"S
    T A T I S T I S C H E{3 SPACE}A U S
    W E R T U N G":LOCATE 15,3:PRINT STR
    ING$(45,"="):LOCATE 23,5:PRINT "Graf
    iken, Trends, Analysen"                      [70F4]
345 ft=1:PLOT 9,395:FOR y=1 TO 2:RESTORE
     3005:FOR x=1 TO 6:READ a,b:DRAWR a*
    ft,b*ft,1:NEXT:ft=-1:NEXT                    [1434]
350 ft=1:PLOT 628,85:FOR y=1 TO 2:RESTOR
    E 3010:FOR x=1 TO 6:READ a,b:DRAWR a
    *ft,b*ft,3:NEXT:ft=-1:NEXT                   [D78E]
355 IF maxeing(0)=0 AND maxeing(1)=0 AND
     hm=0 THEN LOCATE 12,7:PRINT CHR$(16
    4);"{2 SPACE}Klaus Eschemann Am Turm
    sberg 34{2 SPACE}3300 Braunschweig"          [191A]
360 RETURN                                       [CC32]
365 '            *** UP Berechnung und Dars
    tellung Kuchen ***                           [8122]
370 CLS:ueber$=UPPER$(umko$(eingabe)+"/"
    +zeige$+"/"+hm$(hm)):LOCATE 40-INT(L
    EN(ueber$)/2),1:PRINT "{CTRL X} ";ue
    ber$;" {CTRL X}":IF kusum(eingabe) T
    HEN 410                                      [E1FC]
375 lmax=MAX(1,(LEN(maxdm$(eingabe))-3))
    :FOR a=0 TO proko(eingabe)-1:kusu(x)
    =0:FOR b=0 TO anz(eingabe)-1                 [D4FE]
380 kusu(a)=kusu(a)+hwert(eingabe,a,b)/1
    0^lmax:NEXT:kusum(eingabe)=kusum(ein
    gabe)+kusu(a):NEXT                           [E590]
385 kuda(eingabe,0)=(360/kusum(eingabe))
    *kusu(0)                                     [66EE]
390 FOR a=1 TO proko(eingabe):kuda(einga
    be,a)=(360/kusum(eingabe))*(kusu(a))
    +kuda(eingabe,a-1):NEXT                      [A560]
395 hx=320:hy=200:hk=180                         [9C80]
400 FOR a=0 TO proko(eingabe)-1:xw(a)=hk
    *COS(kuda(eingabe,a))+hx:yw(a)=hk*SI
    N(kuda(eingabe,a))+hy:xab(a+1)=xw(a)
    :yab(a+1)=yw(a):IF xab(a)>400 THEN x
    ab(a)=400                                    [3DF2]
405 NEXT:xab(0)=500:yab(0)=400                   [CB10]
410 DEG:FOR a=0 TO 359:PLOT hk*COS(a)+hx
    ,hk*SIN(a)+hy:NEXT a                         [9736]
415 x=proko(eingabe):xw(x)=xw(0)                 [BEAA]
420 FOR a=0 TO proko(eingabe)-1:PLOT hx,
    hy:DRAW xw(a),yw(a):MOVE xw(a)+((xab
    (a)-xw(a))/2)-50,yw(a)+((yab(a)-yw(a
    ))/2)+15:TAG:PRINT proko$(eingabe);a
    +1;:NEXT                                     [8736]
425 RETURN                                       [A036]
430 '           *** UP Darstellung Balken-D
    iagramm ***                                  [3340]
435 t=t(eingabe):anz=anz(eingabe):FOR x=
    0 TO anz-1:mh=mh(eingabe,0,x)                [FB1E]
440 FOR y=0 TO mh-2 STEP 2:MOVE st(einga
    be,x),off+y:DRAWR wt,0,1:NEXT y              [A47A]
445 FOR k=y TO y+tf*sw STEP 2:MOVE st(ei
    ngabe,x)+(k-y)*tg,off+k+1:DRAWR wt,0
    ,3:DRAWR 0,k*-1:NEXT k                       [B9B0]
450 '                  ***-Saeule aufstocken-
    ***                                          [BC10]
455 IF proko(eingabe)<2 THEN 505                 [FE56]
460 FOR sle=1 TO proko(eingabe)-1                [6BF6]
465 FOR y=mh+2 TO mh+mh(eingabe,sle,x)-1
     STEP 2                                      [EA44]
470  MOVE st(eingabe,x),off+y:DRAWR wt,0
    ,3:DRAWR tf*cw,tf*sw,2                       [6628]
475   NEXT y:IF sle=proko(eingabe)-1 THE
    N 495 '                                      [FEF0]
480 FOR k=y TO y+tf*sw STEP 2                    [580E]
485   MOVE st(eingabe,x)+(k-y)*tg,off+k+
    ((mh(eingabe,sle-1,x)+mh(eingabe,sle
    ,x)) MOD 2):DRAWR wt,0,3:DRAWR 0,(y-
    mh)*-1                                       [8394]
490 NEXT k                                       [C30E]
495 MOVE st(eingabe,x),off+y:DRAWR wt,0,
    2                                            [AE94]
500 mh=y:NEXT sle:FOR k=y TO y+tf*sw STE
    P 2:MOVE st(eingabe,x)+(k-y)*tg,off+
    k+2:DRAWR wt,0,3:DRAWR 0,mh*-1:NEXT          [43A4]
505 MOVE st(eingabe,x),off+mh(eingabe,0,
    x):DRAWR wt,0,2:NEXT x:RETURN                [0ACA]
510 '           *** UP Darstellung Trend,B
    erechnung geometr. Trend ***                 [2556]
515 t=t(eingabe):anz=anz(eingabe):lmax=M
    AX(1,(LEN(maxdm$(eingabe))-3))               [E32C]
520 FOR x=1 TO anz-1                             [CA5C]
525 MOVE st(eingabe,x-1)+(wt/2),off+(mh(
    eingabe,pro,x-1)/f):DRAW st(eingabe,
    x)+(wt/2),off+(mh(eingabe,pro,x)/f):
    NEXT x:MOVE xanf,off+(mh(eingabe,pro
    ,0)/f):DRAWR wt/2,0                          [C4B8]
530 '                                            [07BE]
535 diff=ATN(schritt/((mh(eingabe,pro,an
    z-1)/f)-(mh(eingabe,pro,anz-2)/f)+1)
    ):MOVE st(eingabe,anz-1)+(wt/2),off+
    (mh(eingabe,pro,anz-1)/f)                    [A610]
540 DRAW xend,MAX(off+3,off+(mh(eingabe,
    pro,anz-1)/f)+((xend-st(eingabe,anz-
    1))*COS(diff)*SGN(diff)))                    [B5BE]
545 tranz=anz:IF trvor THEN 585                  [3154]
550 FOR x=0 TO anz-1:IF hwert(eingabe,pr
    o,x)<3 THEN hmadm(pro,x)=1:tranz=tra
    nz-1:GOTO 560                                [01F4]
555 hmadm(pro,x)=hwert(eingabe,pro,x)/10
    ^lmax:                                       [D914]
560 NEXT x                                       [1C24]
565 FOR x=0 TO anz-1                             [866C]
570 dm=hmadm(pro,x+1)*hmadm(pro,x):hmadm
    (pro,x+1)=dm:NEXT x:trend=(((hmadm(p
    ro,anz-1)^(1/tranz))/(f(eingabe)*f))
    /t)*10^lmax                                  [0DFA]
575 MOVE st(eingabe,0),off+mh(eingabe,pr
    o,0)/f:DRAW st(eingabe,anz-1)+wt,off
    +(((hmadm(pro,anz-1)^(1/tranz))/(f(e
    ingabe)*f))/t)*10^lmax                       [E8D6]
580 MOVE XPOS,YPOS+6:TAG:PRINT ">";:MOVE
     XPOS-40.YPOS+(20*(-1+(2*eingabe))):
    TAG:PRINT INT(trend*(f(eingabe)*f));
    :GOTO 590                                    [D2F8]
585 yp=YPOS-off:MOVE XPOS-100,YPOS+(60*(
    -1+(2*eingabe))):TAG:PRINT"--->";:PR
    INT USING"######";yp*(f*f(eingabe));
                                                 [1732]
590 MOVE st(eingabe,0)+8,30+(mh(eingabe,
    pro,0)/f)+(40*eingabe):TAG:IF wbep=7
     THEN PRINT"Vorschau ";:IF start=6 T
    HEN PRINT umko$(eingabe);" Total";:G
    OTO 600 ELSE PRINT proko$(eingabe);w
    ;:GOTO 600                                   [E7BA]
595 IF start=6 THEN PRINT umko$(eingabe)
    ;" Total"; ELSE PRINT proko$(eingabe
    );w;                                         [48EE]
600 wbep=0:pro=0:start=0:RETURN                  [E470]
605 '          *** UP Darstellung  B E P
     ***                                         [6BFE]
610 anz=anz(eingabe):t=1:IF t(1-eingabe)
    >t(eingabe) THEN t=t(1-eingabe)              [B2E2]
615 FOR x=1 TO anz-1                             [D866]
620 MOVE st(eingabe,x-1)+(wt/2),(mh(eing
    abe,wbep,x-1)/f)/t+off                       [D788]
625 DRAW st(eingabe,x)+(wt/2),mh(eingabe
    ,wbep,x)/f/t+off:NEXT                        [0D58]
630 MOVE st(eingabe,0),mh(eingabe,wbep,0
    )/f/t+off:DRAWR wt/2,0                       [DBC4]
635 diff=ATN(schritt/((mh(eingabe,wbep,a
    nz-1)/uf(eingabe))-(mh(eingabe,wbep,
    anz-2)/uf(eingabe)))):MOVE st(eingab
    e,anz-1)+(wt/2),mh(eingabe,wbep,anz-
    1)/uf(eingabe)+off                           [E24C]
640 DRAW xend,MAX(off+2,(mh(eingabe,wbep
    ,anz-1)/uf(eingabe))+((xend-st(einga
    be,anz-1))*COS(diff)*SGN(diff))+off)
                                                 [789A]
645 RETURN                                       [B23E]
650 '              *** Steuerung Balken-Diag
    ramm ***                                     [D5B8]
655 CLS:f=f(eingabe):GOSUB 105:GOSUB 435
                                                 [57CE]
660 GOSUB 2285:IF spei THEN GOSUB 2095           [604C]
665 GOTO 1005                                    [23C0]
670 '                *** Steuerung Trend ***
                                                 [BFFE]
675 marg=0:f=f(eingabe):IF trvor THEN 70
    5                                            [44CC]
680 x=start:begr=mh(eingabe,x,0):FOR a=1
     TO anz-1:IF mh(eingabe,x,a)>begr TH
```

```
     EN begr=mh(eingabe,x,a)                  [F734]
685 NEXT:marg=INT(240/begr*10)/10:f=INT(
     (f(eingabe)/marg)*10)/10:IF INT(f)<f
     THEN f=f+(0.5-(f-INT(f)))               [2340]
690 f=MAX(1,f)                                [18B8]
695 GOSUB 105:pro=start:f=f/f(eingabe):G
     OSUB 510:IF spei THEN GOSUB 2095        [439A]
700 CALL &BB18:start=0:GOTO 1005              [A868]
705 '           *** Steuerung Trend Vorsc
     hau ***                                  [A352]
710 pro=7:ueber$=UPPER$(umko$(eingabe)+"
     /VORSCHAU/"+hm$(hm)+umr$(eingabe))      [47D2]
715 begr=mh(eingabe,7,0):FOR a=1 TO anz-
     1:IF mh(eingabe,7,a)>begr THEN begr=
     mh(eingabe,7,a)                          [7872]
720 NEXT:marg=INT(240/begr*10)/10:f=INT(
     (f(eingabe)/marg)*10)/10:IF INT(f)<f
     THEN f=f+(0.5-(f-INT(f)))               [1D2C]
725 f=MAX(1,f)                                [34B6]
730 GOSUB 115:f=f/f(eingabe):GOSUB 510:I
     F spei THEN GOSUB 2095                  [499C]
735 trvor=0:start=0:CALL &BB18:GOTO 1005
                                              [CC40]
740 '            *** Steuerung Break-eve
     n-point ***                              [B62E]
745 CLS:maxdm=maxdm(0,0):eingabe=0:f=f(0
     ):IF maxdm(1,0)>maxdm THEN eingabe=1
     :f=f(1)                                  [C1F4]
750 IF maxdm(eingabe,0)/maxdm(1-eingabe,
     0)>30 THEN MODE 2:LOCATE 10,12:PRINT
     "Gleichzeitige Darstellung der":LOCA
     TE 10,14:PRINT"gewuenschten Umsatz-
     und Kosten-Werte ist nicht moeglich!
     ":LOCATE 10,24:PRINT"Bitte Taste dru
     ecken":GOTO 895                          [6F10]
755 IF trvor THEN GOSUB 155:eingabe=1-ei
     ngabe:GOSUB 155:GOTO 790                [9CD4]
760 '                   ***-Analyse bep-**
     *                                        [13FC]
765 x=start:begr=mh(eingabe,x,0):FOR a=1
     TO anz-1:IF mh(eingabe,x,a)>begr TH
     EN begr=mh(eingabe,x,a)                 [033C]
770 NEXT:marg=280/begr:IF marg>1 THEN f=
     f(eingabe)/marg:f=MAX(ROUND(f),INT(f
     )+0.5)                                   [FC98]
775 ueber$=UPPER$("analyse/"+zeige$+"/"+
     hm$(hm)+umr$):GOSUB 105:wbep=start      [1692]
780 f=f/f(eingabe):uf(eingabe)=f:GOSUB 6
     05                                       [9984]
785 eingabe=1-eingabe:f=f*(f(1-eingabe)/
     f(eingabe)):uf(eingabe)=f:IF t(1-ein
     gabe)>t(eingabe) THEN uf(eingabe)=uf
     (eingabe)*(t(1-eingabe)/t(eingabe))     [7EFE]
786 GOSUB 605:GOTO 820                        [155A]
790 '                   ***-Vorschau bep-*
     **                                       [C1FE]
795 maxdm=mh(0,7,anz-1):eingabe=0:f=f(0)
     :IF mh(1,7,0)>maxdm THEN eingabe=1:f
     =f(1)                                    [7A38]
800 begr=mh(eingabe,7,0):FOR a=1 TO anz-
     1:IF mh(eingabe,7,a)>begr THEN begr=
     mh(eingabe,7,a)                          [8968]
805 NEXT:marg=INT(begr/300*10+0.5)/10:f=
     MAX(INT(f(eingabe)*marg)+1,1)           [6CDA]
810 ueber$=UPPER$("VORSCHAU/"+zeige$+"/"
     +hm$(hm)+umr$(eingabe)):GOSUB 115       [EA0E]
815 f=f/f(eingabe):uf(eingabe)=f:GOSUB 6
     05:eingabe=1-eingabe:f=f*(f(1-eingab
     e)/f(eingabe)):uf(eingabe)=f:IF t(1-
     eingabe)>t(eingabe) THEN uf(eingabe)
     =uf(eingabe)*(t(1-eingabe)/t(eingabe
     ))                                       [9F6C]
816 GOSUB 605                                 [A9F4]
820 '               ***-graf. Ermittlung
     bep-***                                  [570E]
825 IF mh(0,wbep,0)/uf(0)>mh(1,wbep,0)/u
     f(1) AND mh(0,wbep,anz-1)/uf(0)>mh(1
     ,wbep,anz-1)/uf(1) THEN 870             [F2B2]
830 vorzei=1:FOR y=0 TO anz(eingabe)-1:d
     iff=INT((mh(1,wbep,y)/uf(1))-(mh(0,w
     bep,y)/uf(0)))                           [A3E4]
835 IF SGN(diff) <> vorzei AND anzs=0 TH
     EN anzs=y+1                              [B126]
840 NEXT:IF anzs=0 THEN 870                   [E536]
845 anzs=anzs-1:gef=ABS((mh(1,wbep,anzs)
     /uf(1))-(mh(1,wbep,anzs-1)/uf(1)))/s
     chritt:stg=((mh(0,wbep,anzs)/uf(0))-
     (mh(0,wbep,anzs-1)/uf(0)))/schritt      [AB08]
850 spt=0:sp=0:FOR a=st(0,anzs-1) TO st(
     0,anzs):spt=spt+1                        [CC98]
855 top=(mh(0,wbep,anzs-1)/uf(0))+(stg*s
     pt):down=(mh(1,wbep,anzs-1)/uf(1))-(
     gef*spt):abst=down-top                   [85F8]
860 IF abst<=0 AND sp=0 THEN sp=spt           [E712]
865 NEXT a:beph=wt/2+sp+st(0,anzs-1):bep
     v=off+(mh(0,wbep,anzs-1)/uf(0))+(sp*
     stg)                                     [71FE]
870 FOR a=0 TO 1:MOVE st(a,a)+8,off+((mh
     (a,wbep,a)/uf(a))-10)+(30*a):IF wbep
     =7 THEN TAG:PRINT "Vorschau ";          [F77E]
875 IF start=6 THEN TAG:PRINT umko$(a);"
     Total";:GOTO 885                         [82DC]
880 TAG:PRINT proko$(a);w;                    [9CE2]
885 NEXT:IF beph=0 THEN MOVE xend-150,36
     :TAG:PRINT"Kein bep vorhanden";:GOTO
     895                                      [333C]
890 PLOT beph-25,bepv+50:TAG:PRINT INT((
     (bepv-40)*f(eingabe)*uf(eingabe)));:
     MOVE beph,bepv:DRAW beph,40             [343A]
895 wbep=0:trvor=0:start=0:anzs=0:sp=0:b
     eph=0:bepv=0:IF spei THEN GOSUB 2095
                                              [2D34]
900 CALL &BB18:GOTO 1005                      [39C2]
905 '            *** Steuerung Kuchen-Dia
     gramm ***                                [6B20]
910 GOSUB 370:CALL &BB18:IF spei THEN GO
     SUB 2095:GOTO 1005 ELSE GOTO 1005       [0AAE]
915 '              *** Anfangs-Vereinbaru
     ngen ***                                 [10D0]
920 DIM hwert(1,7,23),maxdm(1,23),hmadm(
     6,23),kuda(1,5):bl$="{6 SPACE}":umko
     $(0)="Umsatz":umko$(1)="Kosten":hm$(
     1)="Kuchen-Diagramm":hm$(2)="Balkend
     iagramm":hm$(3)="Trend":hm$(4)="Brea
     k-even-point"                            [A1EC]
925 proko$(0)="Produkt":proko$(1)="Koste
     nart":proko$(2)="Produkte":proko$(3)
     ="Kostenarten":DIM mh(1,7,23),st(1,2
     3):DEFINT e,o,p,y                        [B464]
930 zei$(1)="Jahre":zei$(2)="Jahr":zeig$
     (2)="Monat":zei$(3)="Monate":zei$(4)
     ="Monat":zeig$(4)="Woche":zei$(5)="W
     ochen":zei$(6)="Woche":zeig$(6)="Tag
     ":zei$(7)="Tage":FOR a=1 TO 7 STEP 2
     :zeig$(a)=MID$(zei$(a),1,LEN(zei$(a)
     )-1):NEXT                                [0938]
935 POKE &BDEE,&C9                            [FD28]
1000 '             *** H a u p t m e n u
     e ***                                    [1B40]
1005 CLS:GOSUB 335:LOCATE 72,2:PRINT FRE
     ("")                                     [D7DE]
1010 LOCATE 15,10:PRINT"Kuchendiagramm-D
     arstellung{8 SPACE}======>{3 SPACE}
     1"                                       [BE8E]
1015 LOCATE 15,12:PRINT"Balkendiagramm-D
     arstellung{8 SPACE}======>{3 SPACE}
     2"                                       [A17C]
1020 LOCATE 15,14:PRINT"Trend-Darstellun
     g{17 SPACE}======>{3 SPACE}3"           [C096]
1025 LOCATE 15,16:PRINT"Break-even-point
     -Darstellung{6 SPACE}======>{3 SPAC
     E}4"                                     [981A]
1030 LOCATE 15,18:PRINT"E N D E{27 SPACE
     }======>{3 SPACE}E"                      [AADE]
1035 LOCATE 15,23:PRINT"I H R E{4 SPACE}
     W A H L{4 SPACE}B I T T E{6 SPACE}!
     ":LOCATE 59,23:PRINT"{2 SPACE}":LOC
     ATE 59,23:LINE INPUT a$:a=4:GOSUB 2
     265:IF fehler AND UPPER$(a$)<>"E" T
     HEN fehler=0:GOTO 1035                   [D064]
1040 IF UPPER$(a$)="E" THEN POKE &BDEE,&
     C3:END                                   [A65A]
1045 hm=VAL(a$):ueber$=""                     [634E]
1050 ON hm GOTO 1060,1135,1210,1310           [261E]
1055 '           *** Untermenue Kuchen-D
     iagramm ***                              [CD46]
1060 CLS:GOSUB 335                            [4E7C]
1065 LOCATE 15,8:PRINT"Kuchen-Diagramm":
     LOCATE 15,9:PRINT STRING$(15,"-")       [C2B2]
1070 LOCATE 15,11:PRINT"Eingabe Umsatz{1
     9 SPACE}---->{4 SPACE}1"                 [BF22]
1075 IF maxeing(0) THEN LOCATE 15,11:PRI
     NT"Ansehen"                              [981C]
1080 LOCATE 15,13:PRINT"Eingabe Kosten{1
     9 SPACE}---->{4 SPACE}2"                 [420A]
1085 IF maxeing(1) THEN LOCATE 15,13:PRI
     NT"Ansehen"                              [1824]
1090 LOCATE 15,15:PRINT"NEU-EINGABE (KOR
     REKTUR){10 SPACE}---->{4 SPACE}3"       [79C8]
1095 LOCATE 15,17:PRINT"HAUPTMENUE{23 SP
     ACE}---->{4 SPACE}H"                     [5EC4]
1100 LOCATE 15,23:PRINT"I h r e{4 SPACE}
     W a h l{4 SPACE}B i t t e{5 SPACE}!
     ":LOCATE 57,23:PRINT"{2 SPACE}":LOC
     ATE 57,23:LINE INPUT a$:a=3:GOSUB 2
     265:IF fehler THEN fehler=0:GOTO 11
     00                                       [4346]
1105 IF UPPER$(a$)="H" THEN 1005              [2F34]
1110 um=VAL(a$):IF um=3 THEN 2005             [A558]
1115 eingabe=um-1:IF maxeing(eingabe)>1
     AND proko(eingabe)>1 THEN 905           [FAC6]
1120 IF maxeing(eingabe) THEN MODE 2::LO
     CATE 5,12:PRINT"Fuer Kuchen-Diagram
     m ";umko$(eingabe);"-Werte fuer min
     destens 2 ";proko$(eingabe+2);" ein
     geben!":LOCATE 50,24:PRINT"Weiter d
     urch Tastendruck":CALL &BB18:GOTO 1
     000                                      [69BE]
```

Listing. Umsatz, Kosten, Trends und Analysen (Fortsetzung)



```
1125 GOTO 1445                                    [BC20]
1130 '              *** Untermenue Balken-D
     iagramm ***                                  [CA18]
1135 CLS:GOSUB 335                                [2D82]
1140 LOCATE 15,8:PRINT"Balken-Diagramm":
     LOCATE 15,9:PRINT STRING$(15,"-")            [C884]
1145 LOCATE 15,11:PRINT"Eingabe Umsatz{1
     9 SPACE}---->{4 SPACE}1"                     [2C28]
1150 IF maxeing(0) THEN LOCATE 15,11:PRI
     NT"Ansehen"                                  [2010]
1155 LOCATE 15,13:PRINT"Eingabe Kosten{1
     9 SPACE}---->{4 SPACE}2"                     [4910]
1160 IF maxeing(1) THEN LOCATE 15,13:PRI
     NT"Ansehen"                                  [3218]
1165 LOCATE 15,15:PRINT"NEU-EINGABE (KOR
     REKTUR){10 SPACE}---->{4 SPACE}3"            [78CE]
1170 LOCATE 15,17:PRINT"HAUPTMENUE{23 SP
     ACE}---->{4 SPACE}H"                         [54B8]
1175 LOCATE 15,23:PRINT"I h r e{4 SPACE}
     W a h l{4 SPACE}B i t t e{5 SPACE}!
     ":LOCATE 57,23:PRINT"{2 SPACE}":LOC
     ATE 57,23:LINE INPUT a$:a=3:GOSUB 2
     265:IF fehler THEN fehler=0:GOTO 11
     75                                           [4176]
1180 IF UPPER$(a$)="H" THEN 1005                  [733A]
1185 um=VAL(a$)                                   [608E]
1190 IF um=3 THEN 2005                            [2844]
1195 eingabe=um-1:IF maxeing(eingabe) TH
     EN 650                                       [3960]
1200 GOTO 1445                                    [6814]
1205 '              *** Untermenue Trend ***
                                                  [66A0]
1210 CLS:GOSUB 335                                [3376]
1215 LOCATE 15,7:PRINT"Trend-Darstellung
     ":LOCATE 15,8:PRINT STRING$(17,"-")
                                                  [29B0]
1220 LOCATE 15,9:PRINT"Auswerten Umsatz
     Periode{12 SPACE}---->{4 SPACE}1"            [D700]
1225 IF maxeing(0)=0 THEN LOCATE 15,9:PR
     INT"Eingabe{2 SPACE}"                        [1CF0]
1230 LOCATE 15,11:PRINT"Auswerten Kosten
      Periode{12 SPACE}---->{4 SPACE}2"           [2B36]
1235 IF maxeing(1)=0 THEN LOCATE 15,11:P
     RINT"Eingabe{2 SPACE}"                       [9646]
1240 LOCATE 15,13:PRINT"Vorschau Umsatz
     folgende Periode{4 SPACE}---->{4 SP
     ACE}3"                                       [1A40]
1245 IF maxeing(0)=0 THEN LOCATE 15,13:P
     RINT SPC(32)                                 [8564]
1250 LOCATE 15,15:PRINT"Vorschau Kosten
     folgende Periode{4 SPACE}---->{4 SP
     ACE}4"                                       [0228]
1255 IF maxeing(1)=0 THEN LOCATE 15,15:P
     RINT SPC(32)                                 [AF6C]
1260 LOCATE 15,17:PRINT"NEU-EINGABE (KOR
     REKTUR){13 SPACE}---->{4 SPACE}5"            [328E]
1265 LOCATE 15,19:PRINT"HAUPTMENUE{26 SP
     ACE}---->{4 SPACE}H"                         [6086]
1270 LOCATE 15,23:PRINT"I h r e{4 SPACE}
     W a h l{4 SPACE}B i t t e{5 SPACE}!
     ":LOCATE 60,23:PRINT"{2 SPACE}":LOC
     ATE 60,23:LINE INPUT a$:a=5:GOSUB 2
     265:IF fehler THEN fehler=0:GOTO 12
     70                                           [0F52]
1275 IF UPPER$(a$)="H" THEN 1005                  [DC44]
1280 um=VAL(a$):IF (um=3 OR um=4) THEN t
     rvor=1:IF maxeing(um-3)=0 THEN fehl
     =um-2:GOSUB 2105:fehl=0:trvor=0:GOT
     O 1005                                       [8440]
1285 IF (um=3 OR um=4) THEN trvor=1:eing
     abe=um-3:wbep=7:GOTO 1365                    [4FA4]
1290 IF um=5 THEN 2005                            [004A]
1295 eingabe=um-1:IF maxeing(eingabe) TH
     EN 1365                                      [97CA]
1300 GOTO 1445                                    [9016]
1305 '              *** Untermenue Break-eve
     n-point ***                                  [9FD6]
1310 GOSUB 2105:IF fehl THEN fehl=0:GOTO
      1005                                        [7B52]
1315 CLS:GOSUB 335                                [2F82]
1320 LOCATE 15,9:PRINT"Break-even-point"
     :LOCATE 15,10:PRINT STRING$(16,"-")
                                                  [91CE]
1325 LOCATE 15,12:PRINT"Auswerten vergan
     gene Periode{11 SPACE}---->{4 SPACE
     }1"                                          [9B5C]
1330 LOCATE 15,14:PRINT"Vorschau folgend
     e Periode{14 SPACE}---->{4 SPACE}2"
                                                  [A978]
1335 LOCATE 15,16:PRINT"HAUPTMENUE{29 SP
     ACE}---->{4 SPACE}H"                         [A03C]
1340 LOCATE 15,23:PRINT"I h r e{4 SPACE}
     W a h l{4 SPACE}B i t t e{5 SPACE}!
     ":LOCATE 63,23:PRINT"{2 SPACE}":LOC
     ATE 63,23:LINE INPUT a$:a=2:GOSUB 2
     265:IF fehler THEN fehler=0:GOTO 13
     40                                           [7F50]
1345 IF UPPER$(a$)="H" THEN 1005                  [2F40]
1350 um=VAL(a$):IF um=1 THEN 1365                 [E270]
1355 IF um=2 THEN trvor=1:wbep=7                  [59C8]
1360 '              *** Auswahl Trend/BEP-Einz
     el/Gesamt ***                                [08D2]
1365 CLS:GOSUB 335:LOCATE 15,9:IF hm=3 T
     HEN PRINT umko$(eingabe);" ";                [2D86]
1370 IF trvor THEN PRINT"Vorschau"; ELSE
      PRINT zeige$;                               [654C]
1375 PRINT " ";hm$(hm)                            [2BB6]
1380 LOCATE 15,10:PRINT STRING$(LEN(umko
     $(eingabe))*SGN(hm MOD 4)+LEN(hm$(h
     m))+1+MAX(8*trvor,LEN(zeige$))+SGN(
     hm MOD 4),"-")                               [096E]
1385 LOCATE 15,12:PRINT"Auswahl : ";:IF
     hm=3 THEN PRINT proko$(eingabe);STR
     ING$(10-LEN(proko$(eingabe))," ");"
      (";SGN(proko(eingabe));"bis";proko
     (eingabe);"){7 SPACE}";:GOTO 1395            [8862]
1390 PRINT"Produkt-Kostenart (";SGN(prok
     o(eingabe));"bis";proko(eingabe);")
     ";                                           [D168]
1395 PRINT"{2 SPACE}---->{3 SPACE}1"              [5B0C]
1400 LOCATE 15,14:PRINT "Gesamt{2 SPACE}
     : ";:IF hm=3 THEN PRINT umko$(einga
     be);"{25 SPACE}---->{3 SPACE}2" ELS
     E PRINT"Produkte-Kostenarten";"{11
     SPACE}---->{3 SPACE}2"                       [4648]
1405 LOCATE 15,16:PRINT"Untermenue ";hm$
     (hm);STRING$(20-LEN(hm$(hm))," ");"
     {10 SPACE}---->{3 SPACE}3"                   [9F42]
1410 LOCATE 15,23:PRINT"I h r e{4 SPACE}
     W a h l{4 SPACE}B i t t e{5 SPACE}!
     ":LOCATE 64,23:PRINT"{2 SPACE}":LOC
     ATE 64,23:LINE INPUT a$:a=3:GOSUB 2
     265:IF fehler THEN fehler=0:GOTO 14
     10                                           [294E]
1415 w=VAL(a$):IF w=3 THEN ON hm GOTO 10
     60,1135,1210,1310                            [9280]
1420 IF w=2 THEN start=6:GOTO 1435                [29D8]
1425 LOCATE 15,23:PRINT STRING$(50," "):
     LOCATE 15,23:PRINT"Eingabe ";:IF hm
     =3 THEN PRINT proko$(eingabe);" ?"
     ELSE PRINT"Produkt-Kostenart ?"              [01F4]
1430 LOCATE 64,23:LINE INPUT a$:a=proko(
     eingabe):GOSUB 2265:IF fehler THEN
     fehler=0:GOTO 1430 ELSE w=VAL(a$):s
     tart=w-1                                     [6A2C]
1435 IF trvor THEN GOSUB 155:IF hm=4 THE
     N eingabe=1-eingabe:GOSUB 155                [0D8E]
1440 ON hm GOTO 1000,1000,670,740                 [E864]
1445 '              *** Auswahl Periode *
     **                                           [645C]
1450 IF maxeing(1-eingabe)>0 AND ko=0 TH
     EN prokoz=0:GOTO 1580                        [224C]
1455 CLS:GOSUB 335:LOCATE 15,8:PRINT "Ei
     ngabe ";umko$(eingabe);" / Periode"
                                                  [390A]
1460 LOCATE 15,9:PRINT STRING$(24,"-")            [AA88]
1465 LOCATE 15,10:PRINT"Mehrere Jahre{19
      SPACE}---->{4 SPACE}1"                      [8236]
1470 LOCATE 15,11:PRINT"Ein Jahr{4 SPACE
     }(12 Monate){9 SPACE}---->{4 SPACE}
     2"                                           [8B00]
1475 LOCATE 15,12:PRINT"Mehrere Monate{1
     8 SPACE}---->{4 SPACE}3"                     [70F4]
1480 LOCATE 15,13:PRINT"Ein Monat{3 SPAC
     E}(4 Wochen){10 SPACE}---->{4 SPACE
     }4"                                          [A6A0]
1485 LOCATE 15,14:PRINT"Mehrere Wochen{1
     8 SPACE}---->{4 SPACE}5"                     [DCFE]
1490 LOCATE 15,15:PRINT"Eine Woche{2 SPA
     CE}(7 Tage){12 SPACE}---->{4 SPACE}
     6"                                           [36E2]
1495 LOCATE 15,16:PRINT"Mehrere Tage{20
     SPACE}---->{4 SPACE}7"                       [D8C2]
1500 LOCATE 15,17:PRINT"Untermenue ";hm$
     (hm);STRING$(15-LEN(hm$(hm))," ");"
     {6 SPACE}---->{4 SPACE}8"                    [4E8E]
1505 LOCATE 15,23:PRINT"I h r e{4 SPACE}
     W a h l{4 SPACE}B i t t e{5 SPACE}!
     ":LOCATE 56,23:PRINT"{2 SPACE}":LOC
     ATE 56,23:LINE INPUT a$:a=8:GOSUB 2
     265:IF fehler THEN fehler=0:GOTO 15
     05                                           [DA70]
1510 IF UPPER$(a$)="H" THEN 1005                  [8E34]
1515 pm=VAL(a$):IF pm=8 THEN ON hm GOTO
     1060,1135,1210,1310                          [DC24]
1520 zeige$=zei$(pm)                              [71EE]
1525 IF pm MOD 2=1 THEN LOCATE 15,23:PRI
     NT STRING$(55," "):LOCATE 15,22:PRI
     NT "Ist die Anzahl der ";zei$(pm);"
      bekannt{2 SPACE}?":LOCATE 15,24:PR
     INT"Eingabe: ANZAHL ";UPPER$(zei$(p
     m));"(max 24) ODER ENTER{3 SPACE}";
     :LINE INPUT a$:IF a$="" THEN 1540            [10E8]
1530 a=24:GOSUB 2265:IF fehler THEN fehl
     er=0:GOTO 1525                               [944E]
1535 anz=VAL(a$):anz(eingabe)=anz                 [7BA4]
1540 prokoz=0:LOCATE 15,23:PRINT STRING$
     (50," "):LOCATE 15,24:PRINT STRING$
     (45," "):LOCATE 15,22:PRINT"Ist die
      Anzahl der ";proko$(eingabe+2);" b
     ekannt ?"                                    [5FAA]
1545 LOCATE 15,24:PRINT"Eingabe: ANZAHL(
```

```
     max 6) ODER ENTER{7 SPACE}";:LINE I
     NPUT a$:proko=VAL(a$):IF a$="" THEN
      1560                                [3672]
1550 a=6:GOSUB 2265:IF fehler THEN fehle
     r=0:GOTO 1545                        [23F6]
1555 proko=VAL(a$):IF proko THEN 1585     [26CE]
1560 '            ***-Anzahl Produkte/Kost
     enarten unbekannt-***                [6930]
1565 proko=0:ON pm GOSUB 2140,2150,2140,
     2155,2140,2160,2140,2165             [6B32]
1570 anzz=0:CLS:GOSUB 2210:prokoz=prokoz
     +1:IF hwert(eingabe,prokoz-1,0)>1 A
     ND prokoz<6 THEN 1570                [B108]
1575 proko=prokoz-1-INT(prokoz/7):GOTO 1
     600                                  [6956]
1580 '                                    [DF2A]
1585 '            ***-Anzahl Produkte/Kos
     tenarten bekannt-***                 [10B8]
1590 GOSUB 2170:prokoz=prokoz+1:IF proko
     z<proko THEN 1590                    [8B52]
1595 IF proko(1-eingabe)>proko THEN 1590
                                          [DBDE]
1600 GOSUB 215:maxdm$=STR$(maxdm):dm=INT
     (maxdm/16):dm$=STR$(dm):dm=(INT(dm/
     10^(LEN(dm$)-2))*2)*10^(LEN(dm$)-2)
     :maxdm=dm*16:maxeing(eingabe)=INT((
     (maxdm+(maxdm(eingabe,0)/t))/2/16/1
     0)+1)*10*16:maxdm$(eingabe)=STR$(ma
     xeing(eingabe))                      [4908]
1605 proko(eingabe)=proko:GOSUB 300       [FE10]
1610 ON hm GOTO 905,650,1365,1365         [B89C]
2000 '            *** UP Neu-Eingabe/Korre
     ktur ***                             [3A84]
2005 IF maxeing(0)=0 AND maxeing(1)=0 TH
     EN ON hm GOTO 1060,1135,1210,1310    [37E0]
2010 CLS:GOSUB 335:LOCATE 15,8:PRINT"Neu
     -Eingabe/Korrektur":LOCATE 15,9:PRI
     NT STRING$(21,"-")                   [C152]
2015 LOCATE 15,10:PRINT"Neu-Eingabe tota
     l{16 SPACE}---->{4 SPACE}1"          [274A]
2020 LOCATE 15,12:PRINT"Neu-Eingabe Umsa
     tz{15 SPACE}---->{4 SPACE}2"         [33C8]
2025 LOCATE 15,14:PRINT"Neu-Eingabe Kost
     en{15 SPACE}---->{4 SPACE}3"         [F7B8]
2030 LOCATE 15,16:PRINT"Einzelkorrektur{
     18 SPACE}---->{4 SPACE}4"            [BEEE]
2035 LOCATE 15,18:PRINT"Untermenue ";hm$
     (hm);SPC(22-LEN(hm$(hm)));"---->{4
     SPACE}5"                             [B3C4]
2040 LOCATE 15,23:PRINT"I h r e{4 SPACE}
     W a h l{4 SPACE}B i t t e{5 SPACE}!
     ":LOCATE 57,23:PRINT"{2 SPACE}":LOC
     ATE 57,23:LINE INPUT a$:a=5:GOSUB 2
     265:IF fehler THEN fehler=0:GOTO 20
     40                                   [335A]
2045 ko=VAL(a$):IF ko=5 THEN 2085         [3B5C]
2050 IF ko<>4 THEN 2075                   [6FB6]
2055 LOCATE 15,23:PRINT STRING$(60," "):
     LOCATE 15,23:PRINT"Umsatz (1) oder
     Kosten (2) ?":LOCATE 57,23:INPUT"",
     a$:a=2:GOSUB 2265:IF fehler THEN fe
     hler=0:GOTO 2055 ELSE ko=VAL(a$):ei
     ngabe=ko-1                           [5384]
2060 LOCATE 15,23:PRINT STRING$(60," "):
     LOCATE 15,23:PRINT;proko$(eingabe);
     " - Nummer und ";zeig$(pm);" ?{2 SP
     ACE}(a,b)"                           [66AE]
2065 LOCATE 57,23:INPUT"",p,z:LOCATE 15,
     23:PRINT STRING$(60," "):LOCATE 15,
     23:PRINT"Neuer Wert fuer ";proko$(e
     ingabe);p;",";zeig$(pm);z;"{2 SPACE
     }?":LOCATE 57,23:INPUT"",hwert(eing
     abe,p-1,z-1)                         [6606]
2070 GOSUB 300:GOSUB 215:GOTO 2085        [3BE0]
2075 IF ko=3 OR ko=2 THEN eingabe=ko-2:k
     usum(eingabe)=0:maxeing(eingabe)=0:
     t(eingabe)=1:FOR y=0 TO 7:FOR a=0 T
     O anz(eingabe)-1:hwert(eingabe,y,a)
     =0:mh(eingabe,y,a)=0:maxdm(eingabe,
     a)=0:NEXT:NEXT:GOTO 1445             [C36C]
2080 CLEAR                                [F7A2]
2085 ON hm GOTO 1060,1135,1210,1310       [D730]
2090 '            *** UP Bild abspeichern
     ***                                  [8B24]
2095 c$="!"+umko$(eingabe):POKE &BBD1,25
     5:POKE &BBD2,21:SAVE c$,B,&C000,&40
     00                                   [5A1A]
2100 spei=0:CALL &BB18:RETURN             [C948]
2105 '            *** UP Falschwahl ***   [A01A]
2110 IF maxeing(0)<=1 THEN fehl=1 ELSE I
     F maxeing(1)<=1 THEN fehl=2          [1B9A]
2115 IF fehl THEN CLS ELSE GOTO 2130      [4520]
2120 LOCATE 10,12:PRINT"Fuer ";:IF trvor
      THEN PRINT"Vorschau ";              [BF62]
2125 PRINT hm$(hm);" bitte erst fehlende
     ";umko$(fehl-1);"-Werte eingeben !
     ":LOCATE 6,25:PRINT"Weiter durch Ta
     stendruck":CALL &BB18                [F07A]
2130 RETURN                               [B28C]
2135 '            *** UP DATA-Zuweisung Pe
     riode ***                            [2AA6]
2140 RESTORE 3030:IF anz>0 THEN lesz=anz
     :RETURN                              [02BC]
2145 lesz=24:RETURN                       [FECE]
2150 RESTORE 3015:lesz=12:anz=12:RETURN   [9894]
2155 RESTORE 3020:lesz=4:anz=4:RETURN     [A1DA]
2160 RESTORE 3025:lesz=7:anz=7:RETURN     [48E8]
2165 RESTORE 3030:lesz=24:anz=24:RETURN   [96A6]
2170 '            *** UP Input Umsatz/K
     osten ***                            [1412]
2175 ON pm GOSUB 2140,2150,2140,2155,214
     0,2160,2140,2165:anz(eingabe)=anz:C
     LS                                   [D20C]
2180 WINDOW#1,1,80,25,25:PRINT#1," {CTRL
      X} Kein";CHR$(32+69*eingabe);SPC(e
     ingabe);umko$(eingabe);" angefallen
     :Eingabe=(1) {CTRL X} "              [9D6A]
2185 FOR y=0 TO anz-1                     [1FCE]
2190 READ z$:PRINT"Eingabe ";umko$(einga
     be);" fuer ";proko$(eingabe);prokoz
     +1;"in DM, ";zeig$(pm);" = ";UPPER$
     (z$);" :";:LINE INPUT a$             [64AC]
2195 a=9E+37:GOSUB 2265:IF fehler THEN f
     ehler=0:GOTO 2190 ELSE hwert(eingab
     e,prokoz,y)=VAL(a$):hwert(eingabe,6
     ,y)=hwert(eingabe,6,y)+hwert(eingab
     e,prokoz,y):NEXT                     [F9CE]
2200 IF hwert(1-eingabe,0,0)=0 AND anz(e
     ingabe)=0 THEN CLS#1:PRINT#1,"{CTRL
      X} Fertige Grafik abspeichern{2 SP
     ACE}---> CTRL+A druecken!{3 SPACE}(
     Weiter mit Tastendruck) {CTRL X}":C
     ALL &BB18                            [40A6]
2205 ko=0:a=0:RETURN                      [CFA4]
2210 WINDOW#1,1,80,25,25:PRINT#1,"{CTRL
     X} ";eingabe$(eingabe);" ";zeig$(pm
     );anzz+1;"= 0 ? --->Eingabe = 1 {CT
     RL X} ";"{CTRL X} Eingabe beenden ?
      --->Eingabe = 0 {CTRL X} ":READ z$
                                          [5384]
2215 PRINT"Eingabe ";umko$(eingabe);" fu
     er ";proko$(eingabe);prokoz+1;"in D
     M, ";zeig$(pm);" = ";UPPER$(z$);" :
     ";prokoz;anzz:LINE INPUT a$          [56C6]
2220 a= 9E+37:GOSUB 2265:IF fehler AND V
     AL(a$)<>0 THEN fehler=0:GOTO 2215 E
     LSE IF fehler THEN fehler=0          [6A88]
2225 IF VAL(a$)>0 THEN hwert(eingabe,pro
     koz,anzz)=VAL(a$):hwert(eingabe,6,a
     nzz)=hwert(eingabe,6,anzz)+hwert(ei
     ngabe,prokoz,anzz):GOTO 2235         [D0C2]
2230 ON pm GOSUB 2140,2150,2140,2155,214
     0,2160,2140,2165:GOTO 2245           [313A]
2235 anzz=anzz+1:IF anzz=lesz THEN ON pm
      GOSUB 2140,2150,2140,2155,2140,216
     0,2140,2165                          [7AEE]
2240 GOTO 2210                            [850C]
2245 IF hwert(1-eingabe,0,0)=0 AND anz(e
     ingabe)=0 THEN CLS#1:PRINT#1,"{CTRL
      X} Fertige Grafik abspeichern{2 SP
     ACE}---> CTRL+A druecken!{3 SPACE}(
     Weiter mit Tastendruck) {CTRL X}":C
     ALL &BB18                            [CAB8]
2250 IF anz(eingabe)=0 THEN anz(eingabe)
     =anzz                                [587C]
2255 ko=0:a=0:RETURN                      [DCAE]
2260 '            *** UP Ueberpruefen Inp
     ut ***                               [ED14]
2265 IF INSTR(a$,CHR$(38)) OR INSTR(a$,C
     HR$(43)) OR INSTR(a$,CHR$(45)) OR I
     NSTR(a$,CHR$(46)) THEN fehler=1:GOT
     O 2280                               [6A50]
2270 IF VAL(a$)=0 AND UPPER$(a$)<>"E" AN
     D UPPER$(a$)<>"H" THEN fehler=1      [349C]
2275 IF VAL(a$)<0 OR VAL(a$)>a THEN fehl
     er=1                                 [1CA0]
2280 RETURN                               [9F98]
2285 '            *** UP Abfrage Abspeich
     ern ***                              [FE4A]
2290 a$=INKEY$:IF a$="" THEN 2290         [B2FC]
2295 IF a$="{CTRL A}" THEN spei=1         [4F6C]
2300 RETURN                               [7E8A]
3000 '            *** D A T A 'S ***      [A8F4]
3005 DATA 620,0,3,-2,2,-3,0,-300,-2,-3,-
     3,-2                                 [A3C0]
3010 DATA 3,-2,2,-3,0,-68,-2,-3,-3,-2,-6
     20,0                                 [27C8]
3015 DATA Jan,Feb,Mar,Apr,Mai,Jun,Jul,Au
     g,Sep,Okt,Nov,Dez                    [C250]
3020 DATA 1.Woche,2.Woche,3.Woche,4.Woch
     e                                    [E6FA]
3025 DATA Montag,Dienstag,Mittwoch,Donne
     rstag,Freitag,Samstag,Sonntag        [8622]
3030 DATA 1,2,3,4,5,6,7,8,9,10,11,12,13,
     14,15,16,17,18,19,20,21,22,23,24     [93B8]
```

Listing. Umsatz, Kosten, Trends und Analysen (Schluß)



»Taschen«-Rechner

Wenn Ihr CPC eines kann, dann ist es rechnen. Schließlich bedeutet das englische Wort »Computer« nichts anderes als »Rechner«. Aber ohne ein komfortables Programm, wie den »Mathematiker«, ist es mühsam, ihn zur Mathematik zu bekehren.

Die Bezeichnung »Taschenrechner« ist eigentlich tiefgestapelt, da dieses Programm nicht nur alle Funktionen eines normalen programmierbaren Taschenrechners beherrscht, sondern vieles darüber hinaus.

Nach dem Starten des Programms leuchtet in der unteren linken Ecke des Bildschirms eine »Taste« auf. Bei der hellen Umrandung handelt es sich um den Cursor, der sich mit den Cursorsteuertasten steuern läßt. Bewegt man den Cursor zum Beispiel auf die »Taste« mit dem Pi-Zeichen, läßt sich diese durch den Druck auf die Copy-Taste aktivieren. Als Resultat erscheint die Zahl 3,1415... auf dem Bildschirm im hell umrandeten Ergebnisfeld.

1

HYP DEG 1000000000000001

DATA	DEL	F12	F13	F14	F15	AC
X	NEW	F8	F9	F10	F11	MOD
δn-1	n	F4	F5	F6	F7	PRIM
δn²	Σx	PROG	F1	F2	F3	BINO
δn	Σx²	NOT	X↔M	kgV	ggT	KURZ
arc	HYP	OR	a/b	+/-	ZINS	RAN#
SIN	SIN⁻¹	XOR	1/X	π	%	MR
COS	COS⁻¹	AND	eˣ	ln	P→R	MS
TAN	TAN⁻¹	DEG	√x	x²	R→P	M+
10ˣ	LOG	° ' "	y√x	xʸ	x!	M-

Ihr Bildschirm verwandelt sich in das Bedienerfeld

Zur Erklärung der einzelnen Tastenfunktionen finden Sie im folgenden jeweils die Taste, wie sie auch auf Ihrem Bildschirm erscheint:

10^x Berechnung der Potenz zur Basis 10.
LOG Berechnung des Briggsschen Logarithmus.
o ' " wandelt sexadezimale Zahlen in dezimale um und umgekehrt.
$y\sqrt{x}$ zieht aus der Zahl x die y-te Wurzel.
X^Y erhebt die Zahl x zur y-ten Potenz.
x! berechnet die Fakultät.
M- der angezeigte Wert wird vom Speicherinhalt subtrahiert, bleibt selbst aber unverändert.
TAN berechnet den Tangens der angezeigten Zahl.
TAN^{-1} berechnet den Arcustangens der angezeigten Zahl.
DEG Mit dieser Taste können Sie die Winkelmaße umschalten. Welches Winkelmaß gerade gültig ist, stellt der Computer im Ergebnisfeld durch folgende Kürzel dar:
DEG = Altgrad
RAD = Neugrad
GRA = Bogenmaß
Die trigonometrischen Funktionen berechnen ihre Ergebnisse dem eingestellten Winkelmaß entsprechend.
$\sqrt{x}$ zieht die Wurzel aus der angezeigten Zahl.
X^2 gibt das Quadrat der angezeigten Zahl aus.
R→P wandelt die angezeigte rechtwinklige Zahl in eine polare Zahl um.
M+ der angezeigte Wert wird zum Speicherinhalt addiert, der Wert selbst bleibt unverändert.
COS berechnet den Cosinus der angezeigten Zahl.
COS^{-1} berechnet den Arcuscosinus der angezeigten Zahl.
AND AND-Funktion; bei Aufruf fragt der Rechner nach dem zweiten Wert, der mit dem ersten zu verknüpfen ist.
e^x berechnet den natürlichen Antilogarithmus der angezeigten Zahl: Die Grundzahl »e« (2,7182...) wird zur x-ten Potenz erhoben.
ln berechnet den natürlichen Logarithmus der angezeigten Zahl.
P→R wandelt die angezeigte Zahl in eine rechtwinklige Zahl um.
MS löscht den Speicherinhalt.
SIN berechnet den Sinus der angezeigten Zahl.
SIN^{-1} berechnet den Arcussinus der angezeigten Zahl.
XOR XOR-Funktion (siehe AND)
1/X gibt den Kehrwert der angezeigten Zahl zurück.
π zeigt die Zahl PI an.
% führt Prozentrechnungen durch. Um dem Rechner zu zeigen, was berechnet werden soll, gibt man einfach eine negative Zahl für die gesuchte Größe ein.
MR zeigt den momentanen Speicherinhalt an.
arc rechnet das Winkelargument, von dem angenommen wird, daß es sich auf den in Funktion 10 eingestellten Modus bezieht, in den gewählten Modus um.
HYP Ein Druck dieser Taste verwandelt alle trigonometrischen Funktionen in Hyperbel- oder Areafunktionen. Auf erneuten Druck erhalten die Funktionen ihre auf den Tasten angegebene Bedeutung zurück.
OR OR-Funktion (siehe AND)
a/b rechnet einen Dezimalbruch in eine gemischte Zahl um.
+/- wechselt das Vorzeichen der angezeigten Zahl.
ZINS Zinsrechnung (siehe Prozentrechnung).
RAN# ergibt eine Zufallszahl kleiner 1 aber größer 0.
δ gibt die Grund-Gesamtheitsabweichung der im Statistikspeicher stehenden Zahlen zurück.
Σx^2 gibt die Quadratsumme der im Statistikspeicher stehenden Zahlen aus.
NOT NOT-Funktion (siehe AND).
X↔M vertauscht den angezeigten mit dem Wert im Speicher.
kgV berechnet das kgV (kleinstes gemeinsames Vielfaches) der beiden extra dafür eingegebenen Zahlen.
ggT Gibt den ggT (größten gemeinsamen Teiler) zweier Zahlen zurück.
KURZ kürzt die angezeigte Zahl auf die vorgebene Zahl der Nachkommastellen.
δn^2 berechnet die Varianz der Zahlen im Statistikspeicher.
Σx gibt die Summe der im Statistikspeicher stehenden Zahlen zurück.
PROG fragt nach einem Speicher und zeigt dann die erste Zahl dieses Speichers an.
F1
F2 frei programmierbare Funktionen
F3
BINO gibt Binominalkoeffizienten zurück; sucht das k-te Element der n-ten Zahl.

δn-1	gibt die Häufigkeits-Standardabweichung der im Statistikspeicher befindlichen Daten zurück.
n	gibt die Anzahl der Daten im Statistikspeicher zurück.
F4 F5 F6 F7	frei programmierbare Funktionen
PRIM	entscheidet, ob die angezeigte Zahl eine Primzahl ist.
X	gibt den arithmetischen Mittelwert der Daten im Statistikspeicher an.
NEW	löscht den gesamten Statistikspeicher.
F8 F9 F10 F11	frei programmierbare Funktionstasten
MOD	bietet die Möglichkeit, andere Darstellungsarten für Zahlen zu wählen (binär, hexadezimal); die Ergebnisse werden dann sowohl in dezimaler als auch in der gewählten Form im Ergebnisfeld dargestellt.
DATA	dient zur Dateneingabe in den Statistikspeicher.
DEL	Korrektur der Daten im Statistikspeicher.
F12 F13 F14 F15	frei programmierbare Funktionen
AC	startet das Programm neu (ist als letzte Rettung gedacht, falls das Programm fehlerhaft sein sollte); alle Daten gehen verloren.

Um dem Computer einen neuen Wert zu übermitteln, drückt man nur zweimal die ESCAPE-Taste. Der Computer meldet sich dann mit einem Fragezeichen und erwartet eine Eingabe. Zahlen größer 32768 sind binär nicht darstellbar, werden aber vom Rechner verarbeitet.

Zur Programmierung des Taschenrechners suchen Sie sich eine freie Zeilennummer. Sie sollte möglichst am Ende des Programmes liegen. In diese Zeile schreiben Sie Ihre Funktion. Die Variable »d« beinhaltet immer den Wert, der auf dem Bildschirm zu sehen ist. Sollten Sie als Ergebnisvariable nicht »d« verwenden, ist nicht gewährleistet, daß die nächste Funktion des Programms auch richtig funktioniert (ihr fehlt ja der Wert für »d«). Die Variablen des Hauptprogramms sollten Sie, um Fehler zu vermeiden, nicht wiederverwenden (es sei denn, Sie wollen den Wert der Variablen in einem Unterprogramm einsetzen). Ist die Funktion programmiert, listen Sie die Zeile 2110 auf. Hier fügen Sie, der Nummer der Funktionstaste entsprechend (1 bis 15), die Zeilennummer Ihres Unterprogramms ein. Achten Sie darauf, daß die Position der Zeilennummer in der Zahlenkolonne mit der Nummer auf der Taste übereinstimmt. Anschließend listen Sie Zeile 580 und fügen hier die Zeilennummer des Unterprogramms – an entsprechender Stelle in der Zahlenkolonne – ein. Diesmal unterscheidet sich die entsprechende Stelle von der Nummer der Funktionstaste. Um die richtige Position herauszufinden, müssen Sie in der Tastatur-Abbildung nachsehen, welche Nummer die von Ihnen ausgewählte Taste hat (sie sind von unten links nach oben rechts von 1 bis 70 numeriert). Achtung! Hier ist nicht die Funktionstastennummer (1 bis 15) gemeint. Ist die richtige Nummer herausgesucht, subtrahieren Sie von ihr den Wert 33. Sie suchen nun die der neuen Zahl entsprechende Stelle in der Zahlenkolonne. Die Zahl, die an dieser Stelle in der Kolonne steht, muß auf jeden Fall eine 10 sein, es sei denn, Sie haben diese 10 schon früher ersetzt. Jetzt ersetzen Sie sie durch die Zeilennummer, in der Ihr Unterprogramm beginnt. Sie können jetzt das Programm speichern. Ihre Funktion steht Ihnen damit jedesmal bei Benutzung des Programms zur Verfügung.

Sie wollen beispielsweise die Funktionstaste »F1« programmieren. Dann gehen Sie folgendermaßen vor:

–Zeilennummer suchen:
10000

–Funktion programmieren:
10000 d=d+1:goto 280

–Zeile 2110 ändern:
2110 ON speicher GOTO **10000**,30000,30000,...,...,

–Zeile 580 listen:
580 ...:ON c GOTO.......

–Nummer der Taste heraussuchen:
46

–rechnen:
46-33=13

–Stelle in Zeile 580 suchen:
ON c GOTO 1930,...,...,**10000**,...

Die 10000 muß an 13ter Stelle die 10 ersetzen.

–speichern

Sollte eine Rechnung den Rechenbereich des Computers verlassen, fügt er selbständig eine Fehlermeldung ein. Am Ende des Unterprogramms kehren Sie dann mit dem GOTO-Befehl in die Zeile 280 zurück, da hier die Routine mit der Cursorsteuerung beginnt.

(Jochen Schüler/ja)

Programm-Steckbrief	
Name:	Mathematiker
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
10 REM **********************************
                                        [5EDE]
20 REM          Mathematiker            [E384]
30 REM               by                 [8EC4]
40 REM      Jochen Schueler (C) 1986    [6D4C]
50 REM          Herzogstr. 49           [7BD8]
60 REM        5650 Solingen 11          [38F6]
70 REM **********************************
                                        [ECEA]
80 CLEAR:wert=1:des=1:q=1:f=0:w=0:d=0:KE
   Y DEF 17,1,159:KEY 159,"mode 2:ink 1,
   26:pen 1:ink 0,0:paper 0"            [AD20]
90 SYMBOL AFTER 0:SYMBOL 130,112,136,136
   ,112:SYMBOL 131,128,128,128:SYMBOL 13
   2,160,160,160:SYMBOL 96,0,0,0,136,80,
   32,80,136:SYMBOL 126,1,3,5,113,1,1:SY
   MBOL 35,0,0,0,48,8,4,2,1:SYMBOL 36,1,
   2,4,8,16,32,64,128                   [9FDE]
100 MODE 1:INK 0,0:INK 1,0:INK 2,0:INK 3
    ,0:BORDER 0:PAPER 0:PEN 0:ORIGIN 0,0
    ,1,638,1,400:MOVE 310,48:DRAW 340,48
    ,1:MOVE 320,18:DRAW 350,18:MOVE 5,39
    8:DRAWR 627,0:DRAWR 0,-98:DRAWR -627
    ,0:DRAWR 0,98:WINDOW #1,2,39,2,5    [B1BA]
110 FOR a=47 TO 623 STEP 90:FOR b=14 TO
    284 STEP 30:GOSUB 510               [8A74]
120 NEXT:NEXT                           [D0D6]
130 RESTORE 140:PAPER 0:PEN 1:TAG:FOR i=
    1 TO 30:READ posx,posy,num,a:PLOT po
    sx,posy,1:PRINT CHR$(num);:NEXT     [544A]
140 DATA 16,226,179,1,32,226,110,1,48,22
    6,45,1                              [AE10]
150 DATA 64,226,49,1,24,196,179,1,40,196
    ,110,1,56,203,50,1,403,106,184,1,293
    ,48,35,1,309,48,36,1,303,18,35,1,319
    ,18,36,1,130,226,110,1,123,196,190,1
    ,156,51,126,1,154,83,126,1,154,113,1
    26,1,49,166,110,1,32,166,179,1,132,1
    67,120,1,115,167,190,1,150,173,50   [AC5A]
160 DATA 1,300,76,101,1,318,85,96,1,200,
    16,130,1,229,16,131,1,248,16,132,1,3
    10,167,255,1,490,47,243,1,490,77,243
    ,1                                  [B610]
170 SYMBOL AFTER 0:RESTORE 180:TAG:FOR i
    =1 TO 69:READ posx,posy,num$,a:PLOT
    posx,posy,a:PRINT num$;:NEXT        [3BC2]
```

Listing. Mehr als nur ein Taschenrechner

```
180 DATA 574,19,M-,1,574,47,M+,1,574,77,
    MS,1,574,107,MR,1,565,286,AC,1,565,2
    56,MOD,1,484,18,x!,1,384,137,+/-,1,1
    6,286,DATA,1,38,256,X,1,20,106,SIN,1
    ,20,76,COS,1,20,46,TAN,1,116,286,DEL
    ,1,113,18,LOG,1,20,136,arc,1,20,18,1
    0,1,53,24,x,1,392,16,X,1                [A8FE]
190 DATA 409,22,Y,1,292,167,X,1,330,167,
    M,1,296,137,a/b,1,292,107,1/X,1,393,
    46,X,1,409,52,2,1,323,47,x,1,333,17,
    x,1,289,23,y,1,557,227,PRIM,1,117,25
    7,NEW,1,141,197,x,1,110,136,HYP,1,10
    7,47,TAN,1,107,77,COS,1,107,107,SIN,
    1                                       [5334]
200 DATA 207,46,DEG,1,207,76,AND,1,207,1
    06,XOR,1,210,136,OR,1,207,166,NOT,1,
    490,107,%,1,392,77,ln,1,476,47,R,1,5
    09,47,P,1,476,77,P,1,509,77,R,1,468,
    137,ZINS,1,558,137,RAN#,1,558,167,KU
    RZ,1,558,197,BINO,1,477,167,ggT,1,38
    4,167,kgV,1,197,197,PROG,1              [0414]
210 DATA 302,197,F1,1,392,197,F2,1,482,1
    97,F3,1,212,227,F4,1,302,227,F5,1,39
    2,227,F6,1,482,227,F7,1,212,257,F8,1
    ,302,257,F9,1,387,257,F10,1,479,257,
    F11,1,207,287,F12,1,297,287,F13,1,38
    7,287,F14,1,477,287,F15,1               [A45A]
220 WINDOW #1,2,39,2,5:WINDOW #2,2,5,6,6
    :WINDOW #3,6,9,6,6:WINDOW #4,12,15,6
    ,6:WINDOW #5,17,20,6,6:WINDOW #6,22,
    39,6,6:PRINT #5,"DEG":DEG               [7F60]
230 MOVE 600,800:INK 1,26:INK 2,6:INK 3,
    9:SYMBOL AFTER 0                        [E1FA]
240 CLS #1:INPUT #1,"Haben Sie den Buntm
    onitor dann Taste{2 SPACE}(1) oder h
    aben Sie den Gruenmonitor{3 SPACE}da
    nn Taste (2)";i:IF i<1 OR i>2 THEN 2
    40 ELSE IF i=1 THEN INK 2,15 ELSE IN
    K 2,24                                  [A1F0]
250 ON BREAK GOSUB 2120:ON ERROR GOTO 15
    90                                      [294E]
260 DIM f(1000):CLS #1                      [DF74]
270 a=47:b=13:c=1:GOTO 490                  [D774]
280 FOR i=1 TO 100:NEXT:IF w<>d T4EN 163
    0 ELSE 310                              [1FF2]
290 PRINT #6,"!";BIN$(d,16):w=d:GOTO 310
                                            [394C]
300 PRINT #6,"&";HEX$(d,16):w=d             [C35A]
310 IF INKEY(0)=0 THEN 430                  [33AE]
320 IF INKEY(1)=0 THEN 370                  [39B8]
330 IF INKEY(2)=0 THEN 460                  [5EBC]
340 IF INKEY(8)=0 THEN 400                  [8EBE]
350 IF INKEY(9)=0 THEN 540                  [5ACC]
360 GOTO 280                                [6658]
370 GOSUB 510                               [82E0]
380 a=a+90:IF a>623 THEN 390 ELSE 490       [CD04]
390 a=47:GOTO 490                           [57EA]
400 GOSUB 510                               [5BD4]
410 a=a-90:IF a<47 THEN 420 ELSE 490        [528C]
420 a=587:GOTO 490                          [C350]
430 GOSUB 510                               [04DA]
440 b=b+30:IF b>284 THEN 450 ELSE 490       [B3F8]
450 b=14:GOTO 490                           [BCDA]
460 GOSUB 510                               [91E0]
470 b=b-30:IF b<14 THEN 480 ELSE 490        [F692]
480 b=284                                   [67F2]
490 farbe=2:GOSUB 530                       [083C]
500 GOTO 280                                [6850]
510 farbe=3:GOSUB 530                       [7830]
520 RETURN                                  [AA2E]
530 MOVE a-39,b+13:DRAW a+39,b+13,farbe:
    DRAWR 0,-26:DRAWR -78,0:DRAWR 0,26:M
    OVE 600,1000:RETURN                     [15B0]
540 REM                                     [623A]
550 c=INT(7*(b-13)/30)+((a-47)/90)+1        [4A0E]
560 IF c>33 THEN 580 ELSE 570               [9404]
570 ON c GOTO 590,870,1340,610,790,1610,
    690,880,2270,1060,630,640,1510,710,1
    010,2280,1270,800,1660,1500,730,1040
    ,2290,1280,660,780,810,760,1670,1420
    ,1290,1770,770                          [71EA]
580 c=c-33:ON c GOTO 1930,1980,1260,1250
    ,1300,670,2080,1990,1460,1330,1240,2
    100,10,10,10,2170,1320,1230,10,10,10
    ,10,2140,1310,1130,10,10,10,10,1200,
    1210,1140,10,10,10,10,1120              [9F0E]
590 CLS #1:IF d>37 THEN 320                 [011A]
600 d=10^d:GOTO 1110                        [31A0]
610 CLS #1:INPUT #1,"wievielte Wurzel";w
    :IF w=0 THEN 610                        [666E]
620 d=d^(1/w):GOTO 1110                     [9C32]
630 w=2:GOTO 620                            [A694]
640 w=2:GOTO 860                            [F6A2]
650 RUN 150                                 [B7CC]
660 d=1/d:GOTO 1110                         [C2EE]
670 w=d:d=wariable:wariable=w:GOSUB 740     [5A42]
680 PRINT #1,"X=";d;" M=";wariable:GOTO
    280                                     [AC76]
690 wariable=wariable-d:GOSUB 740           [BF20]
700 GOTO 720                                [FC52]
710 wariable=wariable+d:GOSUB 740           [910E]
720 PRINT #1,"neuer Speicherwert:";waria
    ble:GOTO 280                            [D78E]
730 wariable=0:PRINT #2,"":GOTO 720         [8F8C]
740 IF wariable<>0 THEN PRINT #2,"M" ELS
    E IF wariable=0 THEN PRINT #2," "       [CFAC]
750 RETURN                                  [BE38]
760 CLS #1:PRINT #1,wariable:GOTO 280       [91DC]
770 d=-d:GOTO 1110                          [C48C]
780 d=PI:GOTO 1110                          [6D9E]
790 CLS #1:INPUT #1,"y=",w:GOTO 860         [759C]
800 d=EXP(d):GOTO 1110                      [1BA2]
810 CLS #1:INPUT #1,"Prozentzahl, Grundw
    ert, Prozentwert";p,g,w:IF p<0 THEN
    i=1 ELSE IF g<0 THEN i=2 ELSE IF w<0
     THEN i=3                               [009A]
820 ON i GOTO 850,830,840                   [3C0A]
830 d=(100*w)/p:GOTO 1110                   [93A8]
840 d=(p*g)/100:CLS #1:PRINT #1,d;CHR$(3
    7):GOTO 280                             [7DDC]
850 d=(100*w)/g:GOTO 1110                   [399A]
860 d=d^w:GOTO 1110                         [08DC]
870 d=LOG10(d):GOTO 1110                    [635C]
880 POKE &44,1:GOTO 980                     [9A5C]
890 w=PEEK(&44):IF hyp=1 THEN 910           [3254]
900 ON w GOTO 950,1020,1050                 [DCCA]
910 ON w GOTO 940,930,920                   [FC26]
920 d=0.5*(EXP(d)-(EXP(-d)):GOTO 1110       [400C]
930 d=0.5*(EXP(d)+(EXP(-d)):GOTO 1110       [A90A]
940 d=(((EXP(d))-(EXP(-d)))/((EXP(d))+(E
    XP(-d)))):GOTO 1110                     [DFA8]
950 d=TAN(d):GOTO 1110                      [679A]
960 IF des=3 THEN GOSUB 980                 [0294]
970 IF hyp=1 THEN 890 ELSE 1000             [20D0]
980 IF des<>3 THEN 890                      [10D2]
990 d=d/63.6739892:RAD                      [5F34]
1000 w=PEEK(&44):ON w GOTO 950,1020,1050
                                            [A7FE]
1010 POKE &44,2:GOTO 980                    [BEA2]
1020 d=COS(d):GOTO 1110                     [9EE8]
1030 INPUT #1,"Stelle",st                   [FFA8]
1040 POKE &44,3:GOTO 980                    [1CAA]
1050 d=SIN(d):GOTO 1110                     [9FF8]
1060 FOR i=1 TO 100:NEXT:IF des<3 THEN d
     es=des+1 ELSE des=1                    [43B2]
1070 ON des GOTO 1080,1090,1100             [460E]
1080 PRINT #5,"DEG":DEG:GOTO 280            [6ECA]
1090 PRINT #5,"RAD":RAD:GOTO 280            [D4E8]
1100 PRINT #5,"GRA":GOTO 280                [DABC]
1110 CLS #1:PRINT #1,d:GOTO 280             [2662]
1120 CLS #1:w=0:d=0:PRINT #1,"d:";:INPUT
     #1,"",d:GOTO 280                       [96A4]
1130 s=0:anzahli=0:h=0:qw=0:t=0:anzahl=0
     :ERASE f:DIM f(1000):GOTO 280          [7674]
1140 INPUT #1,"Nr. des gesuchten Data";a
     nzahl:IF anzahl<=0 OR anzahl>100 TH
     EN 1150 ELSE CLS #1:PRINT #1,"das i
     st Data Nr.";anzahl:PRINT #1,"Wert:
     ";f(anzahl):INPUT #1,"neuer Wert";f
     (anzahl):GOTO 280                      [FDCE]
1150 CLS #1:IF anzahl<=0 THEN 1160 ELSE
     PRINT #1,"Datenwerte mit Nr."          [8BD2]
1160 PRINT #1,"es sind keine Daten verfu
     egbar":GOTO 280                        [115A]
1170 FOR i=1 TO anzahli:PRINT #1,"f(i);i
                                            [5DBC]
1180 IF INKEY$="" THEN 1180 ELSE NEXT       [113E]
1190 GOTO 280                               [7EBC]
1200 CLS #1:INPUT #1,"Binaer(1), Hexadez
     imal(2)";put:IF put<1 OR put>2 THEN
      1200 ELSE 280                         [9B2E]
1210 IF anzahl<=0 THEN anzahl=1 ELSE IF
     anzahl<>0 THEN anzahl=anzahl+1:anza
     hli=anzahl                             [B46A]
1220 CLS #1:INPUT #1,"Data";y:f(anzahl)=
     y:t=y*y:qw=qw+t:h=h+y:GOTO 280         [DB82]
1230 CLS #1:INPUT #1,"Ziffernanzahl:",an
     zahl:GOTO 280                          [529A]
1240 CLS #1:PRINT #1,"Datensumme:";h:GOT
     O 280                                  [750A]
1250 CLS #1:PRINT #1,"Quadratsumme:";qw:
     GOTO 280                               [39D8]
1260 d=SQR((qw-(h*h)/anzahl)/anzahl):GOT
     O 1110                                 [8D64]
1270 CLS #1:INPUT #1,w:d=d AND w:GOTO 11
     10                                     [D880]
1280 CLS #1:INPUT #1,w:d=d XOR w:GOTO 11
     10                                     [2CCE]
1290 CLS #1:INPUT #1,w:d=d OR w:GOTO 111
     0                                      [0820]
1300 CLS #1:INPUT #1,w:d=d NOT w:GOTO 11
     10                                     [A0B0]
1310 d=h/anzahl:GOTO 1110                   [B6E2]
1320 d=SQR((qw-(h*h)/anzahl)/(anzahl-1))
     :GOTO 1110                             [1CBC]
1330 d=qw/(anzahl-1):GOTO 1110              [9044]
1340 SYMBOL 130,112,136,136,112:SYMBOL 1
     31,128,128,128:SYMBOL 132,160,160,1
     60:INPUT #1,"Sexagesimal(1) od. Dez
     imal(2)";w:IF w=1 THEN 1350 ELSE IF
      w=2 THEN 1410 ELSE 1340               [B9E0]
1350 CLS #1:IF d<0 THEN PRINT #1,"-";       [4980]
```

```
1360 d=ABS(d):c=FIX(d):PRINT #1,c;CHR$(1
     30)                                   [5CEE]
1370 s=d-c:s=s*60:c=FIX(s)                 [F502]
1380 PRINT #1,c;CHR$(131)                  [C63C]
1390 p=s-c:p=p*60:PRINT #1,p;CHR$(132)     [89FC]
1400 GOTO 280                              [77B0]
1410 CLS #1:INPUT #1,"Stunden, Minuten,
     Sekunden";st,nin,sec:d=(((nin*60)+s
     ec)/3600)+st:GOTO 1110                [07C6]
1420 FOR i=1 TO 100:NEXT:IF hyp=1 THEN 1
     440                                   [CA24]
1430 hyp=1:PRINT #3,"HYP":GOTO 280         [97E4]
1440 hyp=0:PRINT #3,"":GOTO 280            [1C02]
1450 PRINT #4,"DEG":DEG:GOTO 280           [0CCA]
1460 INPUT #1,"auf die wievielte Stelle"
     ;stelle:IF stelle<0 THEN 1460         [7A80]
1470 d=ROUND(d,stelle):GOTO 1110           [75AA]
1480 d=(10^d):GOTO 1110                    [66B0]
1490 d=d^(1/2.71828182):GOTO 1110          [5CBA]
1500 CLS #1:PRINT #1,CHR$(181);:INPUT #1
     ,":",p:INPUT #1,"r:",r:d1=COS(p)*r:
     d2=SIN(p)*r:PRINT #1,"x=";d1:PRINT
     #1,"y=";d2:GOTO 320                   [C4E4]
1510 CLS #1:INPUT #1,"x:",x:INPUT #1,"y:
     ",y:d1=SQR(x^2+y^2):d2=ATN(y/x):PRI
     NT #1,"r=";d1:PRINT #1,CHR$(181)+"=
     ";d2:GOTO 320                         [0DB2]
1520 CLS #1:i=1:zahl=0:w=1:j=1             [7EAC]
1530 PRINT #1,USING "#.";i;:PRINT #1," A
     bordnung";:INPUT #1,f(i):zahl=zahl+
     f(i):IF f(i)=0 THEN 1560              [5F62]
1540 s=1:FOR w=1 TO f(i):s=s*w:NEXT:f(i)
     =s:j=j*f(i):i=i+1                     [905E]
1550 GOTO 1530                             [C21A]
1560 s=1:FOR w=1 TO zahl:s=s*w:NEXT:zahl
     =s:d=zahl/j:GOTO 1110                 [CD1C]
1570 GOTO 1570                             [6C26]
1580 d=0:PRINT #1,"Ergebniszahl nicht me
     hr erfassbar!":RESUME 280             [8512]
1590 IF ERR=8 THEN CLS #1:PRINT #1,"Spei
     cher nicht belegt":RESUME 280:ELSE
     PRINT #1,"Fehler!":RESUME 280         [9EBA]
1600 IF d=0 THEN INPUT #1,"Zahl:",d:d=AB
     S(d)                                  [9C12]
1610 GOSUB 2220                            [6A9C]
1620 GOTO 1110                             [C10A]
1630 IF ((d>32767 OR d<-32768) AND put=1
     ) OR ((d>268435455 OR d<-32769) AND
      put=2) THEN 1631 ELSE 1632           [D0F4]
1631 PRINT #1,"Zahl nicht umwandelbar!":
     CLS #6:w=d:GOTO 310                   [0F04]
1632 IF put=0 THEN put=1                   [6F2E]
1640 ON put GOTO 290,300                   [A7A8]
1650 d=TAN(90-d):GOTO 1110                 [4A22]
1660 d=LOG(d):GOTO 1110                    [B3F6]
1670 pro=0:INPUT #1,"Altgrad(1), Neugrad
     (2), Bogenmass(3)?",sorte:IF sorte<
     1 OR sorte>3 THEN 1670                [88EC]
1680 IF des<=1 THEN GOSUB 1710 ELSE IF d
     es=3 THEN GOSUB 1730 ELSE GOSUB 175
     0                                     [66A0]
1690 IF pro=1 THEN RETURN ELSE GOTO 1110
                                           [9464]
1700 RETURN                                [CC90]
1710 IF sorte=2 THEN d=d*(10/9) ELSE IF
     sorte=3 THEN d=d*(PI/180)             [41E2]
1720 RETURN                                [B094]
1730 IF sorte=1 THEN d=d/(10/9) ELSE IF
     sorte=3 THEN d=d/(10/9)*(PI/180)      [0180]
1740 RETURN                                [8C98]
1750 IF sorte=1 THEN d=d/(PI/180) ELSE I
     F sorte=2 THEN d=d/(PI/180)*(10/9)    [8FB2]
1760 RETURN                                [909C]
1770 zahl=d                                [907E]
1780 ganz=zahl-FIX(zahl)                   [CB40]
1790 gwert=FIX(zahl)                       [E17C]
1800 zaehler=ganz:nenner=1                 [D2DE]
1810 zaehler=zaehler*10:nenner=nenner*10
                                           [2D2C]
1820 ganz1=zaehler-FIX(zaehler)            [5C88]
1830 IF ganz1>0 THEN 1810                  [9746]
1840 z=zaehler                             [0F1E]
1850 n=nenner                              [263E]
1860 r=FIX(z/n):wert=r*n:r=z-wert          [7804]
1870 IF r<1 THEN 1910                      [3B70]
1880 z=n                                   [B32C]
1890 n=r                                   [641E]
1900 GOTO 1860                             [BC24]
1910 PRINT #1,gwert,"(";zaehler/n;"/";ne
     nner/n;")"                            [50A0]
1920 GOTO 280                              [98BE]
1930 INPUT #1,"Zinsen, Kapital, Zinssatz
     , Zeit(Jahre)",z,k,p,t:IF z<0 THEN
     1940 ELSE IF k<0 THEN 1950 ELSE IF
     p<0 THEN 1960 ELSE IF t<0 THEN 1970
      ELSE 1930                            [0924]
1940 d=(k*p*t)/100:GOTO 1110               [6432]
1950 d=(z*100)/(p*t):GOTO 1110             [CEF4]
1960 d=(z*100)/(k*t):GOTO 1110             [F2EC]
1970 d=(z*100)/(k*p):GOTO 1110             [C9E6]
1980 d=RND(6):GOTO 1110                    [77A8]
1990 GOSUB 2010                            [48AC]
2000 GOTO 1110                             [B7FC]
2010 INPUT #1,"x=",z:INPUT #1,"y=",n:zae
     hler=z:nenner=n                       [D1EE]
2020 IF n>z THEN r=z:z=n:n=r               [010E]
2030 r=FIX(z/n):wert=r*n:r=z-wert          [91F0]
2040 IF r<1 THEN d=n:RETURN                [2718]
2050 z=n                                   [5B18]
2060 n=r                                   [620A]
2070 GOTO 2030                             [720E]
2080 GOSUB 2010                            [BF9A]
2090 d=(zaehler*nenner)/n:GOTO 1110        [F8D6]
2100 INPUT #1,"welcher Programmspeicher
     (1-15)";speicher:IF speicher<1 OR s
     peicher>15 THEN 2100                  [35DE]
2110 ON speicher GOTO 30000,30000,30000,
     30000,30000,30000,30000,30000,30000
     ,30000,30000,30000,30000,30000,3000
     0                                     [2024]
2120 CLS #1:w=0:d=0:PRINT #1,"d:";:INPUT
      #1,"",d                              [594C]
2130 RETURN                                [B28C]
2140 IF d=2 THEN 2150 ELSE r=SQR(d):IF d
     /2=FIX(d/2) THEN 2160 ELSE FOR i=3
     TO r STEP 2:IF d/i=FIX(d/i)THEN 216
     0 ELSE NEXT                           [E2CA]
2150 PRINT #1,d;"ist eine Primzahl":GOTO
      320                                  [F330]
2160 PRINT #1,d;"ist keine Primzahl":GOT
     O 320                                 [F708]
2170 INPUT #1,"n,k",n,k                    [5F28]
2180 nk=n-k:d=nk:GOSUB 2220                [99B6]
2190 nk=d:d=k:GOSUB 2220                   [3C98]
2200 k=d:d=n:GOSUB 2220                    [A0B2]
2210 n=d:d=n/(nk*k):GOTO 1110              [B802]
2220 s=1:FOR w=1 TO d:s=s*w:NEXT:d=s:RET
     URN                                   [9726]
2230 INPUT y                               [E520]
2240 x0=LOG(y)                             [7CF2]
2250 coshyp=(EXP(x0)+EXP(-x0)/2):sinhyp=
     (EXP(x0)-EXP(-x0)/2):x1=x0-((coshyp
     -y)/sinhyp)                           [BD32]
2260 PRINT x1:x0=x1:GOTO 2250              [73C8]
2270 POKE &44,1:GOTO 2300                  [66FA]
2280 POKE &44,2:GOTO 2300                  [C0FE]
2290 POKE &44,3                            [A952]
2300 REM                                   [7B92]
2310 IF hyp=0 THEN 2330                    [7614]
2320 w=PEEK(&44):ON w GOTO 2340,2350,236
     0                                     [5578]
2330 w=PEEK(&44):ON w GOTO 2370,2400,243
     0                                     [0C74]
2340 d=0.5*LOG((1+d)/(1-d)):GOTO 1110      [C246]
2350 IF d<1 THEN PRINT #1,"Fehler":GOTO
     320:ELSE d=LOG(d+SQR(d^2-1)):GOTO 1
     110                                   [B53A]
2360 d=LOG(d+SQR(d^2+1)):GOTO 1110         [FE76]
2370 RAD:GOSUB 2600                        [93CA]
2380 d=ATN(d):des=2:GOSUB 1680             [8168]
2390 des=des1:GOTO 1110                    [E154]
2400 IF d=0 THEN d=PI/2:GOTO 1110 ELSE I
     F d=1 THEN d=0:GOTO 1110 ELSE GOSUB
      2480                                 [5E3A]
2410 GOTO 1110                             [AB06]
2420 IF d<0 THEN d=-(180*x)/PI ELSE d=(1
     80*x)/PI:GOTO 1110                    [8458]
2430 IF d>1 THEN PRINT #1,"Fehler":d=0:G
     OTO 1110 ELSE IF d=0 THEN d=0:GOTO
     1110 ELSE IF d=1 THEN d=0:GOTO 1110
      ELSE GOSUB 2480                      [E6B0]
2440 GOTO 1110                             [B40C]
2450 IF d>1 THEN PRINT #1,"Fehler":d=0:G
     OTO 1110:ELSE IF d=0 THEN 1110:ELSE
      IF d=1 THEN d=PI/2:GOTO 1110:ELSE
     GOSUB 2480                            [D21C]
2460 IF d<0 THEN d=-(180*x)/PI ELSE d=(1
     80*x)/PI:GOSUB 1680                   [F406]
2470 GOTO 280                              [75C0]
2480 RAD:GOSUB 2600                        [9CCE]
2490 y=d:a1=0:b1=PI/2                      [EC8C]
2500 FOR i=1 TO 34                         [621E]
2510 x=(a1+b1)/2:IF w=3 THEN f=SIN(x)-y
     ELSE f=y-COS(x)                       [AF74]
2520 IF f<0 THEN a1=x ELSE b1=x            [5FA2]
2530 NEXT                                  [F652]
2540 d=(180*x)/PI                          [6CC0]
2550 des=1:GOSUB 1680                      [D17E]
2560 des=des1                              [A9A6]
2570 RETURN                                [AE9C]
2580 des=des1                              [F1AA]
2590 d=((EXP(d)-EXP(-d))/((EXP(d)+(EXP(-
     d)):GOTO 1110                         [98D4]
2600 pro=1:IF des=0 THEN des=1             [DFA4]
2610 des1=des:IF des=2 THEN sorte=3 ELSE
      IF des=3 THEN sorte=2 ELSE sorte=1
                                           [7784]
2620 RETURN                                [8F94]
```

Listing. Mehr als nur ein Taschenrechner (Schluß)

Ergänzen Sie jetzt Ihre HAPPY-COMPUTER-Sammlung

Schaffen Sie sich ein interessantes Nachschlagewerk und gleichzeitig ein wertvolles Archiv!

Kennen Sie alle »Happy Computer«-Ausgaben von 1985? Suchen Sie einen ganz bestimmten Testbericht? Oder haben Sie einen Teil eines interessanten Kurses versäumt? Suchen Sie nach einer speziellen Anwendung?

Damit Sie jetzt fehlende Hefte mit »Ihrem« Artikel nachbestellen können, finden Sie auf diesen Seiten eine Zusammenstellung aller wesentlichen Artikel der Ausgaben 01 bis 06 und 08 bis 12/85.

Und so kommen Sie schnell an die noch lieferbaren Ausgaben: Prüfen Sie, welche Ausgabe in Ihrer Sammlung noch fehlt, oder welches Thema Sie interessiert. Tragen Sie die Nummer dieser Ausgabe und das Erscheinungsjahr (z.B. 2/85) auf dem Bestellabschnitt der hier eingehefteten Bestell-Zahlkarte ein. Die ausgefüllte Zahlkarte einfach heraustrennen und Rechnungsbetrag beim nächsten Postamt einzahlen. Ihre Bestellung wird nach Zahlungseingang umgehend zur Auslieferung gebracht.

Stichwort	Titel	Seite/Ausgabe
	Aktuelles	
Computer	Amiga — ein Traumcomputer wird Wirklichkeit	9/10
	Atari: Lage gefestigt	14/11
	Der »Plus/4« ist endlich da	12/2
	Grundstein einer neuen Linie und kein zweiter PC	13/10
	Konsequentes Chaos (Der deutsche QL)	14/10
DFÜ	Akustikkoppler für C 64	9/1
	Ascom-Koppler jetzt auch für Atari	20/8
	Ein Anschluß unter dieser Nummer (Mailbox Nummern)	159/3
	Mailboxbetrieb in den USA	22/10
	Neues DFÜ-Programm für den Spectrum	22/10
	Nullmodem zum Aufstecken	12/1
Software	Atari-Schreiber jetzt für 520 ST	14/12
	Software fast zum Nulltarif	10/1
	Träume werden wahr (Schneider-Neuheiten aus England)	9/12
	Wordstar für 199 Mark	5/9
Drucker	Mac Inker, der sparsame Drucker	12/12
Floppy	Commodore-Floppy auf Trab gebracht	9/1
	Diskettenlaufwerk für den Sharp MZ-800	12/1
	Quick Disk — Die Floppy-Alternative (MSX)	20/4
Erweiterung	Mini-Expansion-Box für TI 99/4A	11/1
MSX	Das Musikwunder (Yamaha CX-5)	141/2
	Der Billig-MSX von Philips kommt	50/1
	CP/M mit MSX-Computer: so geht's	141/2
	Ein komplettes System von Philips	19/8
	Flotter Dreier (Sanyo, Goldstar und Canon)	23/5
	MSX-Mix	45/3
	Mit dem fliegenden Teppich auf Erfolgskurs	15/10
Bücher	Bücher für den C 64	71/9
	Bücher zur DFÜ	111/3
	Bücher zu Logo	158/9
	Bücher zum Denken (KI)	120/10
	Messeberichte	
	Computer-Messe Köln: nach wie vor regional	13/9
	Die neuesten Heimcomputer (Winter-CES)	9/3
	Funkausstellung in Berlin: MSX war Trumpf	9/11
	Hacker, Krimis und Spione (Sommer-CES 1985 — Teil 2)	9/9
	Kampf der Kolosse (Winter-CES — Teil 1)	9/4
	Sommer-CES 1985: Weiche Welle in Chicago — Teil 1	9/8
	Software-Jackpot (Winter CES — Teil 2)	9/5
	Software-Super-Show in London (PCW-Show)	12/11
KI	Künstliche Intelligenz in Wiesbaden (AI Europa)	13/12
Musik	Musikmesse Frankfurt: Midi marschiert	22/5
	Interviews	
	David Crane (Ghostbusters Autor)	17/5
	David Snider: Der Grafik-Großmeister	14/9
	Interview mit den »Print Shop«-Machern	14/8
	Jack Tramiel (Chairman Atari)	11/2
	Kay Nishi (Vize-Präsident Microsoft)	120/9
	Hardware-Tests	
Drucker	Bewußt robust (Europrint K 6311 FT)	31/5
	Software-Tests	
Textverarb.	Ein Textprogramm, das sich lohnt (Homeword/C 64)	77/4
	Jane kontra Appleworks	143/9
	Drei Drucker im Test (STX 80, Gemini 10X, CP-80X)	16/1
	(Nachhall auf Seite 149 in 4/85) DWX 305: Schönschrift zum Niedrigpreis	18/2
	Eine heiße Verbindung (EP 22, EP 44, EXD 10)	26/5
	Kompakt und leise: Matrixdrucker GLP (Centronics)	24/1
	Regenbogenfarben — wie gedruckt (Okimate 20)	154/10
	Schön oder schnell (Horizon HX 80)	21/3
	Schöne Schrift mit schnellen Nadeln (Vergleichstest Star SR-10, Epson GX-80, Panasonic KX-1091)	137/9
	Spectrum mit starken Typen (Gabriele 9009)	126/11
	Zwei Drucker für den Schneider (NLQ 401, GP 500 CPC)	112/8
Computer	Chinese mit britischem Paß (Triton 64)	22/2
	Computer der dritten Art — 520 ST und C 128	22/9
	Der Musik Maestro (Yamaha CX-5)	28/4
	Der Neue: Commodore PC 128	46/5
	Der »neue« Spectrum	31/1
	Ein »Einsteiger« aus Taiwan (BIT-90)	16/2
	Joyce — Schneiders Einstieg in die Welt der PCs	24/11
	Quantensprung im Schneckentempo (QL dt. Version)	180/11
	Koreaner mit Deutsch-Talent (Ce-Tec/MSX)	18/3
	Schneiders neue Dimension (CPC 6128)	24/10
	Sharps Jüngster (Sharp MZ-800)	20/1
	Spectrum plus oder Spectrum minus	24/4
	SVI-X'Press — ein starkes Stück gut im Griff (MSX)	128/9
	TO7/70 und MO5E — zwei Computer, ein Konzept	133/9
	Viel Computer für wenig Geld (Schneider CPC 664)	113/8
	YC-64: Fernöstlicher Biedermann (MSX Computer)	20/2
	Wer ist wer? (Atari 520 ST+ und 260 ST)	16/12
	Wie musikalisch ist mein Heimcomputer?	148/11
	9 MSX Computer im Vergleich	124/9
Laufwerke	3-Zoll-Erfahrungen (MCD-1-Floppy für Spectrum)	22/1
	(Discovery/Spectrum)	
	Ein ungleiches Paar (Spectrum — VIC 1541 Interface)	21/4
	Lauf, Floppy, lauf! (SpeedDos plus/C64)	45/12
	Preiswertes Spectr um Floppysystem (Viscount System)	21/2
	Spectrum Diskettensystem im Plus-Look	20/3
	VC 1541 wird zur Rennfloppy	42/4
Recorder	Der Spectrum Sprinter (Datenrecorder: Sprint)	28/1
	Ein billiger Speicher für alle (Recorder MC 3810)	30/5
DFÜ	DFÜ auch mit dem TI (RS 232 für TI 99/4A)	25/5
	Kommunikation mit dem Spectrum	32/4
	Spartanisch aber gut (Ascom Akustikkoppler)	158/3
Sonstiges	Computer steuert Modelleisenbahn	176/11
	Der andere Weg (Spectrum Tastatur)	19/3
	Faszination der Technik (Fischer Technik Roboter)	44/11
	Famose Formel für den C64 (Formel 64)	40/12
	Flachbildschirm mit Schwächen (LCD für Apple IIc)	137/9
	Grafpad Supergrafik für den Spectrum	16/3
	Haltet den Dieb (Alarmanlage für C 64, VC 20)	29/1
	Ohren oder Tasten? (Voice Command Modul/C 64)	40/10
	Peripherie für MSX (Plotter, 3½-Zoll-Floppy) (Joysticks im Vergleichstest)	26/1
	Roboter, Technologie der Zukunft (Fischertechnik)	45/4
	Starker Arm für Heimcomputer (Teach Robot)	38/4
	Tafelfreuden für Grafik-Gourmets (Atari Maltafel)	14/1/
	Vom Piepmatz zum Mini-Orchester (Spectrum Sound)	15/2
	Software-Tests	
Textverarb.	Ein Textprogramm, das sich lohnt (Homeword/C 64)	77/4
	Jane Kontra Appleworks	143/9
	Jedem seine Zeitung (The Newsroom)	118/8
	Schreiben ohne Frust	46/2
	Star Texter: die 3-Sterne-Textverarbeitung (CPC 464)	45/9
	Textverarbeitung für jedermann (Homewriter für MSX)	137/1
Sprachen	Basic-Erweiterung zum Spartarif (Aztec Basic/C 64)	76/4

Stichwort	Titel	Seite/Ausgabe
	Drei Assembler für Atari-Computer im Vergleich	30/3
	Fortschritt rückwärts (CP/M-80 Emulator für 520 ST)	138/11
	Hisoft-Pascal jetzt Microdrive-kompatibel (Spectrum)	56/2
	Logo für den Atari 520 ST	134/11
	Mallard-80-Basic — ein starkes Stück	28/11
	Maschinensprache ist keine Zauberei (CPC 464)	107/8
	Personal-Basic für den Atari 520 ST	27/9
	Prozessor-Welt von morgen: C 64 simuliert 68000	42/10
	Spezielles Spiele-Basic für den Spectrum	143/5
	Welches Basic für meinen MZ-700?	48/2
	Zwölf Farben in Mode 2 (Color Star für CPC 464)	110/8
Utilities	Das Programm, das Programme macht (Progressor)	33/5
	Disketten-Doktor für den C 128	42/12
	Quicksave für Spectrum	137/4
	SM-Kit — Das Werkzeug für Lehrling und Meister (C 64)	138/1
	Software-Knackern dazwischengepfuscht (Apple II)	27/3
Grafik	Beeindruckend (Print Shop — Druckprogramm)	50/2
	Die Maus bringt Farbe auf den Bildschirm (Apple)	52/2
	Grafik grandios (Malprogramm Blazing Paddles)	28/3
	Koala Bilder zum Anfassen (Hardcopy-Programm)	57/2
	Mit dem Joystick programmiert (Designers Pencil)	140/5
	Schneiders Künstleratelier (Grafikmaster)	43/9
	Viel Grafik für wenig Geld (Graphics Basic und Supergrafik 64 für C 64 im Vergleich)	44/2
	Vorsicht Kamera! (Take 1, Trickfilm Designer)	126/8
DFÜ	Apple II sucht Anschluß	154/3
	Contact 64 — Die Software zum Ascom-Koppler	142/5
	Spectrum auf Draht (DFÜ Vergleichstest)	124/8
Astronomie	Spectrums Sternstunden	34/3
	Sterngucker	158/10
Schach	Schachmatt per Telefon	156/10
	Spiele-Tests	
	Amazon	145/5
	Amazon	145/9
	Archon II: Adept	126/2
	Asylum	144/3
	Athletic Land	146/1
	A View to a Kill	169/10
	Ballblazer	167/10
	Blade of Blackpool	146/9
	Boulder Dash	125/2
	Bounty Bob strikes back	139/8
	Cavelord	124/2
	Crazy Train	144/1
	D-Bug	118/2
	Deus ex Machina	146/4
	Don't buy this	168/12
	Doomdark's Revenche	148/5
	Dorodon	142/3
	Dragonsden	124/2
	Dragonworld	146/9
	Drop Zone	150/9
	Elektro Freddy	145/1
	Elite	164/10
	Eureka	144/4
	Fahrenheit 451	145/5
	Five-a-Side Football	166/10
	Formula One	140/8
	Frank Brunos Boxing	166/10
	Frankie goes to Hollywood	162/10
	Fruity Frank	145/4
	Ghettoblaster	169/11
	Ghostbusters	138/3
	Ghost Chaser	170/11
	Great American Cross Country Road Race	168/11
	Gremlins	148/9
	Hacker	167/12
	H.E.R.O.	149/5
	Hyper Sports	149/9
	Hyper Sports 1	143/3
	Jump Jet	148/9
	Karateka	146/4
	Kennedy Approach	168/12
	Knight Lore	143/3
	Macbeth	144/4
	Mask of the Sun	122/2
	Mask of the Sun	145/9
	Match Day	150/5
	Mindshadow	141/8
	Mr. Do	167/10
	Monster Trivia	168/10
	Nick Faldo plays the Open	169/11
	Nightshade	169/12
	Nodes of Yesod	169/12
	On Court Tennis	150/5
	Pitfall II	148/5
	Rama	145/5
	Rescue on Fractalus	168/10
	Rocket Ball	140/8
	Rockford's Riot (Bolder Dash II)	168/11
	Rock'n Bolt	139/8
	Sherlock Homes	121/2
	Seastalker	147/1
	Serpent's Star	142/4
	Shadowfire	151/9
	Six-Gun Shootout	149/9
	Software Star	165/11
	Spelunker	142/3
	Standing Stones	145/4
	Starion	151/9
	Summer Games II	133/8
	Super Pipeline II	141/8
	The Ancient Art of War	149/5
	The Dallas Quest	147/9
	The Dam Buster	150/9
	The Fourth Protocol	165/11
	The Hitchhiker's Guide to the Galaxy	138/4
	The Hitchhiker's Guide to the Galaxy	147/9
	The Hobbit	146/9
	The Little Computer People Projekt	170/12
	The Way of exploding Fist	169/10
	Tour de France	170/11
	Where in the World is Carmen San Diego	163/11
	Whistler's Brother	141/3
	White Lightning	148/1
	Winter Games	164/12
	World Championship Boxing	170/12
	ZimSalaBim	141/3
	Spiele Tips	
	Abenteuer im Weltraum	152/5
	Alien 8	154/9

Stichwort	Titel	Seite/Ausgabe
	Amazon	
	Atlantis	
	Aztec Challenge	
	Aztec Tomb	
	Aztec Tomb	
	Beach Head	
	Blade of Blackpool	
	Castle of Terror	
	Critical Mass	
	Dallas Quest	
	Death in the Caribbean	
	Death in the Caribbean	
	Death in the Caribbean	
	Doomdark's Revenge	
	Eureka	
	Everyone's a Wally	
	Forbidden Forest	
	Forest at World's End	
	Ghostbusters	
	Ghostbusters	
	Ghostbusters	
	Hampstead	
	Heros of Karn	
	Heros of Karn	
	Hexenküche	
	Hobbit	
	Hulk	
	Hunch Back	
	Jewels of Babylon	
	Karateka	
	Lode Runner	
	Lode Runner	
	Mask of the Sun	
	Masquerade	
	Message from Andromeda	
	Mindshadow	
	Miner 2049er	
	Pirate Adventure	
	Pitfall	
	Pitfall II	
	Pitfall II	
	Sabre Wulf	
	Sands of Egypt	
	Sands of Egypt	
	Schloß des Grauens	
	Secret Mission	
	Ship of Doom	
	Sorcerer of Claymorgue Castle	
	Spelunker	
	Strip Poker	
	Summer Games	
	Summer Games	
	Super Huey	
	The Dallas Quest	
	The Hitchhiker's Guide to the Galaxy	
	The Institute	
	The Institute	
	The Quest	
	The Witness	
	Time Maschine	
	Transsylvanien	
	Ultima II	
	Ultima II	
	Ultima III	
	Ulysses	
	Valhalla	
	Whistler's Brother	
	Zeppelin	
	ZimSalaBim	
	Zork	
	Listings	
Anwendung	Alle Neune (Jahresauswertung-Kegeln/C 64)	
	(Adreßverwaltung/C 64) Besseres Basic ganz einfach Software Basic 3.0/C 64)	
	Datenbank mit freiem Zugriff (C 64)	
	Der Halleysche Komet kommt (MSX)	
	Der Spectrum am Telefon (L.d.M./Spectrum)	
	Die Mini-Textverarbeitung (Spectrum)	
	Nachhall auf Seite 160 in 9/85	
	Do-it-yourself-Datenverwaltung (Mainfile II/C 64)	
	Einblick ins Innenleben (Disassembler/CPC 464)	
	Eine tolle Textverarbeitung für den Schneider (464)	
	Nachhall auf Seite 85 in 5/85	
	Funktionen optisch aufbereitet (VZ-200/Laser)	
	Geregelte Finanzen mit dem Commodore 64	
	Nachhall auf Seite 117 in 8/85	
	Morse-Decoder für Funkamateure (Spectrum)	
	Nebenkostenabrechnung (C 64)	
	Nachhall auf Seite 80 in 12/85 Optik mit Simons Basic (C 64)	
	Programme in Reih' und Glied (C 64)	
	Suchen, nein danke (Dateiverwaltung/CPC 464)	
	Nachhall auf Seite 117 in 8/85	
	Transistor-Schaltungen berechnen (L.d.M./Spectrum)	
	Turbo-Basic-Interpreter für Atari 800XL (L.d.M.)	
Grafik	Apple IIc-HiRes-Grafik auf dem Drucker	
	Bewegte Bilder auf dem C 64	
	Bewegte Grafik mit drei Befehlen (CPC 464)	
	Bewegung: vom Sprite zum Zeichentrick (C 64)	
	Farbspielereien (Atari)	
	Grafikentzerrung für Matrixdrucker (Spectrum)	
	Grafik-Window bekommt Nachwuchs (C 64)	
	Nachhall auf Seite 80 in 12/85	
	Grafikzauber (Apple II)	
	Hires Fantasy (C 64)	
	Rosetten-Grafik für den Spectrum	
	Schnelle Grafik aus dem Compiler (L.d.M./C 64)	
	Schnelle Sprites auf allen Apple-Computern	
	Schöne schnelle Grafik (Grafik-Paket/C 64)	
	Solar-Painter (Spectrum)	
	Sprite-Editor (C 64)	
	Sprites drehen ganz einfach (C 64)	
	Sprites per Software (CPC 464)	
	Vom Bild zum Sprite (C 64)	
	Zauber der Farben mit Magic Painter (L.d.M./Atari)	
	Nachhall auf Seite 85 in 5/85	
	Zaubereien auf dem Bildschirm (L.d.M./Grafik/CPC 464)	
	Zeichenroutine für Kreise und Ellipse (CPC 464)	
	Zykloide für Grafiker und Mathematiker (C 64)	
	Nachhall auf Seite 79 in 12/85	
Spiel	Das Haus des Magiers (C 64)	
	Dasher, der Volltreffer (L.d.M./C 64)	
	Nachhall auf Seite 117 in 8/85	
	Der rasende Raider (C 64)	
	Diamantenfieber (L.d.M./Atari 48 KByte)	
	Nachhall auf Seite 85 in 5/85	
	Die Abenteuer eines rasenden Reporters (Report/C 64)	
	Geröllheimer (Atari)	
	Gespensterjagd im Schneider (CPC 464)	
	Nachhall auf Seite 85 in 5/85	
	Lumberjack Larrys Abenteuer in Bagdad (S.d.M/C 64)	
	Kalte Zeiten (Wintry Screen/C 64)	
	Kneipe zum hastigen Kellner (VÇ 20)	
	Mit dem Apple auf die Trainerbank (Aktion Apfelsaft)	
	Mit dem Atari-Computer auf Ölsuche (Atari)	
	Mit Woodshot ins Manöver ziehen	
	Mücke mit Tücke (C 64)	
	Nachtflug (Spectrum)	
	Nachhall auf Seite 85 in 5/85 Niemandsland (C 64)	
	Pokerface für 16 KByte (Spectrum)	
	Psycho — die Macht des Geistes (C 64)	
	Nachhall auf Seite 80 in 12/85	
	Rennfahrer mit dem Joystick (Driver/C 64)	
	Rettet den letzten Baum (Insekt defense/C 64)	
	SAM — der Mann von der Baustelle (L.d.M./CPC 464)	
	Nachhall auf Seite 79 in 12/85	
	Schatzhöhle (Atari 800XL)	
	Vorsicht Hochwasser (Aquantor/L.d.M./C 64)	
	Über den Wolken (Flugplanung/C 64)	
	Wortsuchspiel (Spectrum)	
Tips&Tricks	AMPEL — grünes Licht für Atari-Maschinen-Programme	
	Auf dem Laufenden mit einer Echtzeituhr (C 64)	
	Auf Trap gebracht (CPC 464)	
	Autostart für Atari	
	Basic bequem (C 64)	
	Basic-Compactor (Spectrum)	
	Nachhall auf Seite 80 in 12/85	
	Basic-Plus: Applesoft-Basic-Erweiterung (Apple II)	
	Beim C 64 piepst es	
	Bilder richtig konservieren (CPC 464)	
	Bildschirmtrick für den Commodore 64	
	Byte-Shifter (Spectrum)	
	Data-Generator für Apple II	
	Dateien hin- und hergerissen (Atari-IBM)	
	Der neue Checksummer ist da (C 64)	
	Der neue Checksummer ist da (C 64)	
	Der neue Checksummer (C 64)	
	Deutsche Sonderzeichen unter CP/M (CPC 464)	
	Die Maltafel wird zur Maus (Atari)	
	Disketten sparen (C 64)	
	Nachhall auf Seite 49 in 11/85	
	Disk-Help für die schnelle Hilfe (Atari)	
	Disk- und DOS-Utility für alle Atari-Computer	
	Drei Tricks für MSX	
	Ein langes Gesicht für den C 64 (Longscreen 64)	
	Fehlerhilfe mit HELP & TRACE (VC 20)	
	Fensterkünstler (C 64)	
	Fettschrift für den 48 KByte-Spectrum	
	Find Label (Spectrum)	
	Flotte Primzahlen in Hisoft Pascal (Spectrum)	
	Funktionstas. mit bel. langen Befehlsfolgen (CPC 464)	

Mathe-Assistent

Bei den Hausaufgaben in der Vektorgeometrie hilft Ihnen jetzt Ihr CPC. Mit dem Programm »Vektor« analysieren Sie problemlos dreidimensionale Vektorräume. Machen Sie Ihren Schneider zum unermüdlichen Übungspartner.

Routinearbeiten aus der Vektorgeometrie, wie zum Beispiel Vektorprodukt, Skalarprodukt, Einheitsvektor, lineare Abhängigkeit etc. führt dieses Programm für Sie aus. Für Schüler ist »Vektor« ein wertvolles Hilfsprogramm zur Prüfung von Aufgabenlösungen.

Die Programmbedienung erfolgt menügesteuert und ist sehr einfach. Sie haben die Auswahl zwischen 9 Menüpunkten (siehe Abbildung). Nach der Auswahl einer Funktion müssen Sie die Koordinaten und eventuell noch die Anzahl der zu verknüpfenden Vektoren eingeben.

Die Programmvariablen werden am Anfang des Listings kommentiert. Abänderungen und Erweiterungen des Programms sind dadurch kein Problem.

(Karl Henes/Matthias Rosin)

```
                 Vektorberechnung
                 ----------------

     Vektorensumme......................1
     Vektorendifferenz..................2
     Kreuzprodukt.......................3
     Skalarprodukt......................4
     Vektorenlinearitaet................5
     Ortsvektor des Teilpunktes.........6
     Vektorbetrag.......................7
     Einheitsvektor.....................8
     Winkel zwischen zwei Vektoren....9


                 Ihre Wahl: _
```

Alle Funktionen auf einen Blick

Programm-Steckbrief	
Name:	Vektor
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
10 REM ************************           [5442]
20 REM *  Vektorberechnung  *             [96EC]
30 REM * copyright K. Henes *             [2A40]
40 REM ************************           [9C48]
50 REM * Variablendefinition *            [E7B4]
60 ' V(10,3) * Vektorkoordinatenfeld *    [5138]
70 ' a$      * Hilfsvariable         *    [7CC6]
80 ' t       * Laufvariable          *    [1B4A]
90 ' Anzahl  * Vektoranzahl          *    [8636]
100 ' X       * X-Koordinate          *   [1DF2]
110 ' Y       * Y-Koordinate          *   [D6F8]
120 ' Z       * Z-Koordinate          *   [CFFE]
130 DIM V(10,3) ' * Vektorfeld (maximal
    10 dreidimensionale Vektoren) *       [6C56]
140 CLS                                   [752E]
150 INK 1,0:INK 0,13:BORDER 13:MODE 2 '
    * Farben *                            [8E6E]
160 REM * Rahmen zeichnen *               [773C]
170 MOVE 9,9:DRAW 9,391:DRAW 631,391:DRA
    W 631,9:DRAW 9,9:MOVE 14,14:DRAW 14,
    386:DRAW 626,386:DRAW 626,14:DRAW 14
    ,14:LOCATE 1,1:PRINT CHR$(129);:LOCA
    TE 80,1:PRINT CHR$(130);:LOCATE 80,2
    5:PRINT CHR$(136);:LOCATE 1,25:PRINT
     CHR$(132);                           [F484]
180 REM * Menue *                         [F296]
190 LOCATE 32,3:PRINT"Vektorberechnung"   [1DFA]
200 LOCATE 32,4:PRINT"----------------"   [7E54]
210 LOCATE 23,7:PRINT"Vektorensumme.....
    ..................1"                  [E238]
220 LOCATE 23,8:PRINT"Vektorendifferenz.
    ..................2"                  [40FA]
230 LOCATE 23,9:PRINT"Kreuzprodukt......
    ..................3"                  [ADEA]
240 LOCATE 23,10:PRINT"Skalarprodukt....
    ..................4"                  [867C]
250 LOCATE 23,11:PRINT"Vektorenlinearita
    et...............5"                   [0E4C]
260 LOCATE 23,12:PRINT"Ortsvektor des Te
    ilpunktes.......6"                    [04BC]
270 LOCATE 23,13:PRINT"Vektorbetrag.....
    ..................7"                  [0FFC]
280 LOCATE 23,14:PRINT"Einheitsvektor...
    ..................8"                  [2D12]
290 LOCATE 23,15:PRINT"Winkel zwischen z
    wei Vektoren....9"                    [FB90]
300 LOCATE 34,19:PRINT"Ihre Wahl:  _"     [F800]
310 REM * Keyabfrage mit Fehlermeldung *
                                          [0912]
320 a$=INKEY$:IF a$="" THEN 320 ELSE IF
    ASC(a$)<49 OR ASC(a$)>57 THEN LOCATE
     16,22:PRINT "Nur eine ganze Zahl zw
    ischen 1 und 9 eingeben !":PRINT CHR
    $(7);:FOR t=1 TO 2000:NEXT:LOCATE 16
    ,22:PRINT SPACE$(47):GOTO 320         [4D02]
330 LOCATE 45,19:PRINT a$:ON ASC(a$)-48
    GOTO 340,440,540,630,730,830,980,108
    0,1180 ' * Verzweigung *              [0AE2]
340 REM * Berechnung der Vektorensumme *
                                          [AC80]
350 CLS:BORDER 1                          [AEC6]
360 LOCATE 2,2:PRINT "Vektorensumme":LOC
    ATE 2,3:PRINT "-------------"         [6BB0]
370 LOCATE 10,5:INPUT;"Eingabe der Vekto
    renanzahl (maximal 10) - ",anzahl:an
    zahl=INT(anzahl):IF anzahl<1 OR anza
    hl>10 THEN SOUND 1,1000,50:GOTO 370   [DDEC]
380 LOCATE 10,8:PRINT "Eingabe der Vekto
    rkoordinaten - ":FOR T=1 TO ANZAHL:L
    OCATE 9,T+9:PRINT T;". VEKTOR":LOCAT
    E 25,T+9:INPUT;"X:  ",V(T,1):LOCATE 4
    0,T+9:INPUT;"Y:  ",V(T,2):LOCATE 55,T
    +9:INPUT;"Z:  ",V(T,3):NEXT           [97F0]
390 FOR T=1 TO ANZAHL:X=X+V(T,1):Y=Y+V(T
    ,2):Z=Z+V(T,3):NEXT                   [724C]
400 LOCATE 10,22:PRINT "Summenvektor - "
    : LOCATE 35,22:PRINT "X-Koordinate >
     ";X:LOCATE 35,23:PRINT "Y-Koordinat
    e > ";Y:LOCATE 35,24:PRINT "Z-Koordi
    nate > ";Z                            [B1BE]
410 WHILE INKEY$="":WEND                  [C2B6]
420 IF INKEY(59)=0 THEN 340               [E82E]
430 GOTO 140                              [714A]
440 REM * Berechnung der Vektorendiffere
    nz *                                  [13AE]
450 CLS:BORDER 1                          [5FC8]
460 LOCATE 2,2:PRINT "Vektorendifferenz"
    :LOCATE 2,3:PRINT "-----------------
    "                                     [7946]
470 LOCATE 10,5:INPUT;"Eingabe der Vekto
    renanzahl (maximal 10) - ",anzahl:an
    zahl=INT(anzahl):IF anzahl<1 OR anza
    hl>10 THEN SOUND 1,1000,50:GOTO 370   [8DEE]
480 LOCATE 10,8:PRINT "Eingabe der Vekto
    rkoordinaten - ":FOR T=1 TO ANZAHL:L
    OCATE 9,T+9:PRINT T;". VEKTOR":LOCAT
    E 25,T+9:INPUT;"X:  ",V(T,1):LOCATE 4
    0,T+9:INPUT;"Y:  ",V(T,2):LOCATE 55,T
    +9:INPUT;"Z:  ",V(T,3):NEXT           [7EF2]
490 X=V(1,1):Y=V(1,2):Z=V(1,3):FOR T=2 T
    O ANZAHL:X=X-V(T,1):Y=Y-V(T,2):Z=Z-V
    (T,3):NEXT                            [2B80]
500 LOCATE 10,22:PRINT "Differenzvektor
    - ": LOCATE 35,22:PRINT "X-Koordinat
```

```
    e > ";X:LOCATE 35,23:PRINT "Y-Koordi
    nate > ";Y:LOCATE 35,24:PRINT "Z-Koo
    rdinate > ";Z                          [6310]
510 WHILE INKEY$="":WEND                   [69B8]
520 IF INKEY(59)=0 THEN 440                [B432]
530 GOTO 140                               [BE4C]
540 REM * Berechnung des Kreuzproduktes
    *                                      [D480]
550 CLS:BORDER 1                           [C0CA]
560 LOCATE 2,2:PRINT "Kreuzprodukt":LOCA
    TE 2,3:PRINT "-------------"           [EAA4]
570 LOCATE 10,8:PRINT "Eingabe der Vekto
    rkoordinaten - ":FOR T=1 TO 2:LOCATE
     9,T+9:PRINT T;". VEKTOR":LOCATE 25,
    T+9:INPUT;"X: ",V(T,1):LOCATE 40,T+9
    :INPUT;"Y: ",V(T,2):LOCATE 55,T+9:IN
    PUT;"Z: ",V(T,3):NEXT                  [06DA]
580 X=V(1,2)*V(2,3)-V(1,3)*V(2,2):Y=V(1,
    3)*V(2,1)-V(1,1)*V(2,3):Z=V(1,1)*V(2
    ,2)-V(1,2)*V(2,1)                      [E508]
590 LOCATE 10,15:PRINT "Produktvektor -
    ": LOCATE 35,15:PRINT "X-Koordinate
    > ";X:LOCATE 35,16:PRINT "Y-Koordina
    te > ";Y:LOCATE 35,17:PRINT "Z-Koord
    inate > ";Z                            [6BCA]
600 WHILE INKEY$="":WEND                   [34B8]
610 IF INKEY(59)=0 THEN 540                [D934]
620 GOTO 140                               [094C]
630 REM * Berechnung des Skalarproduktes
     *                                     [FE1A]
640 CLS:BORDER 1                           [ABCA]
650 LOCATE 2,2:PRINT "Skalarprodukt":LOC
    ATE 2,3:PRINT "-------------"          [9198]
660 LOCATE 10,5:INPUT;"Eingabe der Vekto
    renanzahl (maximal 10) - ",anzahl:an
    zahl=INT(anzahl):IF anzahl<1 OR anza
    hl>10 THEN SOUND 1,1000,50:GOTO 370   [F8F0]
670 LOCATE 10,8:PRINT "Eingabe der Vekto
    rkoordinaten - ":FOR T=1 TO ANZAHL:L
    OCATE 9,T+9:PRINT T;". VEKTOR":LOCAT
    E 25,T+9:INPUT;"X: ",V(T,1):LOCATE 4
    0,T+9:INPUT;"Y: ",V(T,2):LOCATE 55,T
    +9:INPUT;"Z: ",V(T,3):NEXT             [8CF4]
680 X=V(1,1):Y=V(1,2):Z=V(1,3):FOR T=2 T
    O ANZAHL:X=X*V(T,1):Y=Y*V(T,2):Z=Z*V
    (T,3):NEXT:ERGEBNIS=X+Y+Z              [30BE]
690 LOCATE 10,22:PRINT "Ergebnis > ":LOC
    ATE 23,22:PRINT ERGEBNIS               [033A]
700 WHILE INKEY$="":WEND                   [C3BA]
710 IF INKEY(59)=0 THEN 630                [A836]
720 GOTO 140                               [724E]
730 REM * Berechnung der Vektorenlineari
    taet *                                 [C35C]
740 CLS:BORDER 1                           [60CC]
750 LOCATE 2,2:PRINT "Vektorlinearitaet"
    :LOCATE 2,3:PRINT "-----------------
    "                                      [824E]
760 LOCATE 10,8:PRINT "Eingabe der Vekto
    rkoordinaten - ":FOR T=1 TO 3:LOCATE
     9,T+9:PRINT T;". VEKTOR":LOCATE 25,
    T+9:INPUT;"X: ",V(T,1):LOCATE 40,T+9
    :INPUT;"Y: ",V(T,2):LOCATE 55,T+9:IN
    PUT;"Z: ",V(T,3):NEXT                  [A7DE]
770 ERGEBNIS=V(1,1)*V(2,2)*V(3,3)+V(2,1)
    *V(3,2)*V(1,3)+V(3,1)*V(1,2)*V(2,3)-
    V(3,1)*V(2,2)*V(1,3)-V(1,1)*V(3,2)*V
    (2,3)-V(2,1)*V(1,2)*V(3,3)             [13FA]
780 LOCATE 10,15:PRINT"Ergebnis > ":LOCA
    TE 23,15:PRINT ERGEBNIS                [A802]
790 IF ERGEBNIS=0 THEN LOCATE 10,17:PRIN
    T "Die Vektoren sind linear abhaengi
    g." ELSE LOCATE 10,17:PRINT "Die Vek
    toren sind nicht linear abhaengig."   [C886]
800  WHILE INKEY$="":WEND                  [48FC]
810 IF INKEY(59)=0 THEN 730                [CA3A]
820 GOTO 140                               [FF50]
830 REM * Berechnung vom Ortsvektor des
    Teilpunktes *                          [EA7A]
840 CLS:BORDER 1                           [15CE]
850 LOCATE 2,2:PRINT "Ortsvektor des Tei
    lpunktes":LOCATE 2,3:PRINT "--------
    -------------------"                   [DE0E]
860 LOCATE 10,8:PRINT "Eingabe des Ortsv
    ektors vom Anfangspunkt - "            [77A4]
870 LOCATE 10,10:PRINT "1 . Vektor":LOCA
    TE 25,10:INPUT;"X: ",V(1,1):LOCATE 4
    0,10:INPUT;"Y: ",V(1,2):LOCATE 55,10
    :INPUT;"Z: ",V(1,3)                    [8EEC]
880 LOCATE 10,12:PRINT "Eingabe des Orts
    vektors vom Endpunkt - "               [3CB0]
890 LOCATE 10,14:PRINT "2 . Vektor":LOCA
    TE 25,14:INPUT;"X: ",V(2,1):LOCATE 4
    0,14:INPUT;"Y: ",V(2,2):LOCATE 55,14
    :INPUT;"Z: ",V(2,3)                    [8218]
900 LOCATE 10,16:PRINT "Eingabe von Lamb
    da - "                                 [2638]
910 LOCATE 10,18:INPUT;"Lambda (nicht 0
    oder -1): ",LAMBDA                     [9DB0]
920 IF LAMBDA=0 OR LAMBDA=-1 THEN 910      [873C]
930 V(2,1)=V(2,1)*LAMBDA:V(2,2)=V(2,2)*L
    AMBDA:V(2,3)=V(2,3)*LAMBDA:X=V(1,1)+
    V(2,1):Y=V(1,2)+V(2,2):Z=V(1,3)+V(2,
    3):LAMBDA=LAMBDA+1:X=X/LAMBDA:Y=Y/LA
    MBDA:Z=Z/LAMBDA                        [8494]
940 LOCATE 10,21:PRINT "Ortsvektor des T
    eilpunktes - ": LOCATE 40,22:PRINT "
    X-Koordinate > ";X:LOCATE 40,23:PRIN
    T "Y-Koordinate > ";Y:LOCATE 40,24:P
    RINT "Z-Koordinate > ";Z               [6044]
950 WHILE INKEY$="":WEND                   [EDC8]
960 IF INKEY(59)=0 THEN 830                [E348]
970 GOTO 140                               [025C]
980 REM * Berechnung des Vektorenbetrage
    s *                                    [77E2]
990 CLS:BORDER 1                           [C4DA]
1000 LOCATE 2,2:PRINT "Vektorbetrag":LOC
     ATE 2,3:PRINT "------------"          [C09C]
1010 LOCATE 10,8:PRINT "Eingabe der Vekt
     orkoordinate - "                      [3DD0]
1020 LOCATE 10,10:PRINT "Vektor":LOCATE
     25,10:INPUT;"X: ",V(1,1):LOCATE 40,
     10:INPUT;"Y: ",V(1,2):LOCATE 55,10:
     INPUT;"Z: ",V(1,3)                    [1EF6]
1030 X=V(1,1)*V(1,1):Y=V(1,2)*V(1,2):Z=V
     (1,3)*V(1,3):ERGEBNIS=SQR(X+Y+Z)      [F294]
1040 LOCATE 10,15:PRINT "Ergebnis > ":LO
     CATE 23,15:PRINT ERGEBNIS             [6B8E]
1050 WHILE INKEY$="":WEND                  [DF18]
1060 IF INKEY(59)=0 THEN 980               [70A4]
1070 GOTO 140                              [8DAC]
1080 REM * Berechnung des Einheitsvektor
     s *                                   [E18A]
1090 CLS:BORDER 1                          [1C2A]
1100 LOCATE 2,2:PRINT "Einheitsvektor":L
     OCATE 2,3:PRINT "--------------"      [A91A]
1110 LOCATE 10,8:PRINT "Eingabe der Vekt
     orkoordinate - "                      [C5D2]
1120 LOCATE 10,10:PRINT "Vektor":LOCATE
     25,10:INPUT;"X: ",V(1,1):LOCATE 40,
     10:INPUT;"Y: ",V(1,2):LOCATE 55,10:
     INPUT;"Z: ",V(1,3)                    [0AF8]
1130 X=V(1,1)*V(1,1):Y=V(1,2)*V(1,2):Z=V
     (1,3)*V(1,3):BETRAG=SQR(X+Y+Z):B1=X
     +Y+Z:X=V(1,1)/BETRAG:Y=V(1,2)/BETRA
     G:Z=V(1,3)/BETRAG                     [9F74]
1140 LOCATE 10,14:PRINT "Betrag des Vekt
     ors{2 SPACE}={2 SPACE}Wurzel aus";B
     1;"multipliziert mit x/y/z.":LOCATE
      10,17:PRINT "Einheitsvektor - ": L
     OCATE 35,17:PRINT "X-Koordinate > "
     ;X:LOCATE 35,18:PRINT "Y-Koordinate
      > ";Y:LOCATE 35,19:PRINT "Z-Koordi
     nate > ";Z                            [459C]
1150 WHILE INKEY$="":WEND                  [061A]
1160 IF INKEY(59)=0 THEN 1080              [EFF6]
1170 GOTO 140                              [5AAE]
1180 REM * Berechnung des Winkels zwisch
     en zwei Vektoren *                    [C968]
1190 CLS:BORDER 1:DEG                      [8040]
1200 LOCATE 2,2:PRINT "Winkel zwischen z
     wei Vektoren":LOCATE 2,3:PRINT "---
     --------------------------"           [A23E]
1210 LOCATE 10,8:PRINT "Eingabe des erst
     en Vektors - "                        [D7BE]
1220 LOCATE 10,10:PRINT "1 . Vektor":LOC
     ATE 25,10:INPUT;"X: ",V(1,1):LOCATE
      40,10:INPUT;"Y: ",V(1,2):LOCATE 55
     ,10:INPUT;"Z: ",V(1,3)                [E838]
1230 LOCATE 10,12:PRINT "Eingabe des zwe
     iten Vektors - "                      [FA02]
1240 LOCATE 10,14:PRINT "2 . Vektor":LOC
     ATE 25,14:INPUT;"X: ",V(2,1):LOCATE
      40,14:INPUT;"Y: ",V(2,2):LOCATE 55
     ,14:INPUT;"Z: ",V(2,3)                [DD64]
1250 X=V(1,1):Y=V(1,2):Z=V(1,3):FOR T=2
     TO 2:X=X*V(T,1):Y=Y*V(T,2):Z=Z*V(T,
     3):NEXT                               [98AC]
1260 ERGEBNIS=X+Y+Z:X=V(1,1)*V(1,1):Y=V(
     1,2)*V(1,2):Z=V(1,3)*V(1,3):ERGEBNI
     S1=SQR(X+Y+Z):X=V(2,1)*V(2,1):Y=V(2
     ,2)*V(2,2):Z=V(2,3)*V(2,3):ERGEBNIS
     2=SQR(X+Y+Z)                          [12FE]
1270 ERGEBNIS3=ERGEBNIS1*ERGEBNIS2:ERGEB
     NIS=ERGEBNIS/ERGEBNIS3:IF ERGEBNIS=
     1 THEN ERGEBNIS=0:GOTO 1300           [3D84]
1280 REM * Neu: Berechnung des Arcuscosi
     nus *                                 [C908]
1290 RAD:ERGEBNIS=-ATN(ERGEBNIS/SQR(-ERG
     EBNIS*ERGEBNIS+1))+1.5707633:ERGEBN
     IS=180/PI*ERGEBNIS                    [8932]
1300 LOCATE 10,18:PRINT "Ergebnis > ":LO
     CATE 23,18:PRINT ERGEBNIS;" Grad."    [B32E]
1310 WHILE INKEY$="":WEND                  [5C16]
1320 IF INKEY(59)=0 THEN 1180              [2DF4]
1330 GOTO 140                              [98AA]
1340 END                                   [417E]
```

Listing. Vektorberechnung mit dem Computer

Bildschirm de Luxe

Unterstreichen, Fett- und Kursivschrift und manche Hervorhebung mehr kann der Schneider im 80-Zeichen-Modus. Daß sich die Ausgabegeschwindigkeit dabei noch nahezu verdoppelt, fällt fast gar nicht mehr auf.

Wie Sie wahrscheinlich längst wissen, heißt der Teil im Betriebssystem, der die Zeilenausgabe steuert, »TEXT SCREEN«. Und diese Routine kann sehr viel: Verwalten von acht Bildschirmfenstern, drei Schriftbreiten (Modus 0 bis 2), Ausführen von Steuerzeichen (CHR$(=) bis CHR$(31)), Darstellung des Cursors, Verwalten der Farben, transparenter Schreibmodus und natürlich die Zeichenausgabe auf dem Bildschirm, um nur die wichtigsten aufzuzählen. Es gibt aber auch einige Unzulänglichkeiten, und die sollen natürlich nicht verschwiegen werden.

Da ist einmal die sehr langsame Zeichenausgabe auf dem Bildschirm. Geben Sie einmal das kleine Programm aus Listing 1 ein.

Nach dem Start füllt sich der gesamte Bildschirm mit dem Buchstaben »a« und anschließend wird die dafür benötigte Zeit ausgegeben. Da uns nur die Zeit der reinen Zeichenausgabe interessiert, muß man von der im Programm ermittelten Zeit den Betrag abziehen, den das Programm zur Ausgabe von Leerstrings (a$=" ") benötigt. Diese Zeit erhalten wir, wenn wir Listing 2 laufen lassen. In Zeile 80 wird deshalb der Betrag 0,146666667 abgezogen.

Das Beschreiben des Bildschirms mit Zeichen – es ist übrigens egal, mit welchem Zeichen der Schirm beschrieben wird – dauert also etwa 3.29 Sekunden. Mit dem gleichen Programm messen wir nachher die Zeit, die das Beschreiben des Bildschirms benötigt, nachdem die Routine TEXT SCREEN geändert wurde.

Der Hauptgrund für diesen ziemlich hohen Zeitaufwand liegt bei den drei verschiedenen Schriftbreiten. Die Zeichenausgaberoutine des Betriebssystems muß natürlich so konzipiert sein, daß sie allen drei Modi gerecht wird. Dies beinhaltet dann für den Modus 2 (80 Zeichen) eine Unzahl überflüssiger Berechnungen. Zumindest in diesem Modus kann man die Zeichenausgabe erheblich beschleunigen. Und wenn wir schon eine neue Ausgaberoutine schreiben, dann soll sie auch noch mehr können.

Erweitern wir TEXT SCREEN im Modus 2 also um folgende Funktionen:
- Bildschirmausgabe beschleunigen
- Unterstreichen
- Fettschrift
- Kursivschrift
- Durchstreichen
- Durch-x-en auf dem Bildschirm

Die Modi 0 und 1 sollen unverändert funktionieren.

Die Hervorhebungen (Unterstreichen, Fettschrift und so weiter) werden am geschicktesten mittels Bildschirmsteuerzeichen ein- und ausgeschaltet. Hier bietet sich das noch unbelegte Steuerzeichen CHR$(27) an. Mit einem Parameter, der CHR$(27) folgt, wird die Hervorhebungsart bestimmt und in Flip-Flop-Manier ein- und ausgeschaltet.

Im allgemeinen ruft die Routine TXT OUT ACTION (ein Programmteil des Betriebssystems beziehungsweise des TEXT SCREEN) den TEXT SCREEN auf. Das Programm (zum Beispiel der Basic-Interpreter beim PRINT-Befehl) übergibt im A-Register des Z80 den ASCII-Wert des auszugebenden Zeichens. Taucht hier ein Wert kleiner oder gleich 31 auf, so »weiß« der TEXT SCREEN, daß ein Steuerzeichen vorliegt. Er verzweigt in einen Programmabschnitt, der Steuerzeichen bearbeitet.

Für jedes Steuerzeichen liegt in einer Tabelle (die vom Betriebssystem beim Initialisieren im RAM angelegt wird) ein 3-Byte-Wert vor, der die weitere Verarbeitung bestimmt.

Das erste Byte besagt, wieviele folgende Ausgabezeichen (entsprechend der Zahl der Aufrufe von TXT OUT ACTION) als Parameter des Steuerzeichens zu interpretieren sind. Auch wenn diese Zeichen nun einen ASCII-Wert größer 31 haben, werden sie nicht ausgegeben, sondern bis zum Erreichen der benötigten Gesamtzahl der Parameter zwischengespeichert. Die anderen beiden Byte geben die Adresse der Routine an, die die Funktion des Steuerzeichens ausführt. Ihr Aufruf erfolgt automatisch, sobald der TEXT SCREEN alle benötigten Parameter »gesammelt« hat. Nach dem Ende der Routine werden die folgenden Zeichen – sofern es sich nicht wieder um Steuerzeichen handelt – normal auf dem Bildschirm ausgegeben.

Ursprünglich sind für CHR$(27) 0 Parameter sowie eine direkte Rückkehradresse eingetragen. Mit anderen Worten, CHR$(27) hat keine Funktion. Setzen wir nun in die Tabelle eine 1 sowie die Adresse unseres Programmes ein, kann der folgende Parameter ausgewertet und zum Ein- beziehungsweise Ausschalten von Hervorhebungsarten benutzt werden.

Zusammen mit einer neuen Textausgaberoutine (genauer einer neuen Routine für TXT WRITE CHAR des TEXT SCREEN, die für die Darstellung eines Zeichens auf dem Bildschirm verantwortlich ist), die in Modus 2 sowohl eine Beschleunigung der Bildschirmausgabe bewirkt, sowie für eine Darstellung der jeweils eingeschalteten Hervorhebungsarten sorgt, ist die Erweiterung des TEXT SCREEN komplett.

Das Assemblerprogramm mit diesen Funktionen finden Sie in Listing 3, den entsprechenden Basic-Lader in Listing 4. Nachdem das Programm mit »CALL &A000« initialisiert wurde, steht der erweiterte TEXT SCREEN bis zum Ausschalten des Computers zur Verfügung.

Lassen Sie nun nach der Initialisierung der Erweiterung das obige Zeitmeßprogramm erneut laufen. Die Geschwindigkeitserhöhung ist unmittelbar zu verfolgen. Man benötigt lediglich noch 1.89 Sekunden zum Beschreiben des gesamten Bildschirmes. Das entspricht einer Beschleunigung um etwas mehr als 40 Prozent.

Die einzelnen Hervorhebungsarten werden mit
- PRINT CHR$(27);CHR$(1)
  Unterstreichen ein/aus
- PRINT CHR$(27);CHR$(2)
  Fettschrift ein/aus
- PRINT CHR$(27);CHR$(3)
  Kursivschrift ein/aus
- PRINT CHR$(27);CHR$(4)
  Inverse Darstellung ein/aus
- PRINT CHR$(27);CHR$(5)
  Durchstreichen ein/aus
- PRINT CHR$(27);CHR$(6)
  Überstreichen ein/aus
- PRINT CHR$(27);CHR$(7)
  Durch-x-en ein/aus

ein- beziehungsweise ausgeschaltet.

Die Zeichenfolge CHR$(27);CHR$(x) kann natürlich auch ein String enthalten. Zur Demonstration dient Listing 5.



Es ist auch möglich, mehr als eine Hervorhebungsart gleichzeitig zu aktivieren, so zum Beispiel Fettschrift und Unterstreichen oder jede andere beliebige Kombination. Nur Fett- und Kursivschrift heben sich gegenseitig auf.

Unsere Routine sieht eine Funktion zum Blockieren von Änderungen der Hervorhebungsart vor. Nach der Eingabe von

PRINT CHR$(27);CHR$(0)

werden alle folgenden Änderungskommandos ignoriert. Das heißt, der Zustand, der zu diesem Zeitpunkt vorliegt (beispielsweise Unterstreichen und inverse Darstellung) bleibt erhalten. Um diesen Zustand wieder aufzuheben, ist die Eingabe eines Parameters mit einem Wert größer als 127 nötig, also zum Beispiel

PRINT CHR$(27);CHR$(128)

Nun werden alle Veränderungen der Hervorhebungsart wieder bearbeitet.

Das Assemblerprogramm ist folgendermaßen aufgebaut. Zunächst werden – da man tunlichst nur über die vorgegebenen Vektoren Betriebssystemroutinen aufrufen soll – die entsprechenden Restarts der Vektoren in Aufrufe umgewandelt. Dies funktioniert problemlos, weil die erforderliche Speicherkonfiguration bereits vor dem Aufruf unseres Programms eingestellt ist.

Die Vektoren enthalten einen Restart-Befehl, gefolgt von einer Startadresse im Betriebssystem. Daß bei dieser Adresse das siebte Bit gesetzt ist, hat etwas mit dem angewählten ROM zu tun und braucht uns hier nicht weiter zu interessieren. Der Restart schaltet nun die Speicherkonfiguration um, so daß die in den beiden obersten Bits angewählte ROM-Adresse aktiviert wird. Der Aufruf erfolgt dann im richtigen Speicherbereich. Das ist notwendig, damit die Vektoren von jeder beliebigen Speicherkonfiguration aus funktionsfähig sind.

Da unser Assemblerprogramm, wenn es initialisiert ist, nur vom TEXT SCREEN aus aufgerufen wird, ist immer gewährleistet, daß das untere ROM eingeschaltet ist. Die Umkonfiguration des Speichers mit einem Restart gestaltet sich sehr zeitaufwendig. Ein einfacher Aufruf hingegen ist erheblich schneller.

Das Programm kopiert also die hinter den Restarts der Vektoren stehenden Startadressen (nachdem Bit 7 zurückgesetzt wurde). Somit ist auch die Kompatibilität zu anderen Betriebssystemen (CPC 464, 664 und 6128 haben unterschiedliche Betriebssysteme) gewahrt. Nach dem Kopieren wird der Vektor für TXT WRITE CHAR auf den entsprechenden Startpunkt im Programm gesetzt und die Tabelle für die Steuerzeichen abgeändert. Im folgenden steht die Ersatzroutine für TXT WRITE CHAR und das Programm zur Behandlung der Steuerzeichen.

Die Hervorhebungsarten werden im wesentlichen innerhalb des Assemblerprogramms erklärt. Fettschrift beziehungsweise Kursivschrift entsteht dadurch, daß das Bitmuster der Buchstaben jeweils um ein Bit verschoben und anschließend mit dem Originalmuster OR- (Fettschrift) beziehungsweise AND-verknüpft (Kursivschrift) wird. Dadurch erscheint die Schrift auf dem Bildschirm fett und heller beziehungsweise schlank und dunkler als normal.

(Matthias Engelbach/hg)

Programm-Steckbrief	
Name:	Bildschirm de Luxe
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette, Diskette

```
10 MODE 2                                  [DFF0]
20 a$=STRING$(80,"a")                      [3A92]
30 t=TIME                                  [96C6]
40 FOR z=1 TO 25                           [C97A]
50    LOCATE 1,z                           [E828]
60    PRINT a$                             [C430]
70 NEXT z                                  [99C0]
80 PRINT (TIME-t)/300-0.146666667          [F306]
```

Listing 1. Mehr als drei Sekunden, um den Bildschirm voll zu schreiben

```
10 MODE 2                                  [DFF0]
20 a$=""                                   [8210]
30 t=TIME                                  [96C6]
40 FOR z=1 TO 25                           [C97A]
50    LOCATE 1,z                           [E828]
60    PRINT a$                             [C430]
70 NEXT z                                  [99C0]
80 PRINT (TIME-t)/300                      [4F30]
```

Listing 2. Die Programmbearbeitung ist schnell

```
 10 ;
 20 ;
 30 ;
 40 ;
 50 ;          Bildschirmausgabesteuerung
 60 ;          ==========================
 70 ;
 80 ;
 90 ;
100 ;
110 ;      Funktion: Die Ausgabe von Zeichen in MODE 2
120 ;                wird von diesem Programm erweitert.
130 ;                Neben einer Beschleunigung der Ausgabe
140 ;                um etwa den Faktor 2 werden ueber
150 ;                das Steuerzeichen CHR$(27) + CHR$(x)
160 ;                folgende Ausgabefunktionen realisiert:
170 ;
180 ;                - Unterstreichen
190 ;                - Doppeldruck / Fettschrift
200 ;                - Kursivschrift / Schmalschrift
210 ;                - Inverse Darstellung (zusaetzlich
220 ;                  zu CHR$(24) !)
```

Listing 3. Der Quellcode für »Bildschirm de Luxe«



```
                230 ;           - Durchstreichen
                240 ;           - Ueberstreichen
                250 ;           - Durch - x - en
                260 ;
                270 ;
                280 ;           Die Ausgabe von Zeichen in MODE 0 und
                290 ;           MODE 1 wird nicht beeinflusst.
                300 ;
                310 ;
                320 ;
                330 ;
A000            340         org #A000
                350 ;
                360 ;
                370 ;       Initialisieren
                380 ;
                390 ;       RESTARTs  in  JUMPs  umwandeln
                400 ;
                410 ;
A000  2AA6BB    420         ld   hl,(#BBA6)          ; Vektor von TXT GET MATRIX
A003  CBBC      430         res  7,h                 ; ROM-Bit ruecksetzen
A005  226AA0    440         ld   (getmat),hl         ; in Programm eintragen
                450 ;
A008  2A1BBC    460         ld   hl,(#BC1B)          ; Vektor von SCR CHAR POSITION
A00B  CBBC      470         res  7,h                 ; ROM-Bit ruecksetzen
A00D  2278A0    480         ld   (chrpos),hl         ; in Programm eintragen
                490 ;
A010  2A12BC    500         ld   hl,(#BC12)          ; Vektor von SCR GET MODE
A013  CBBC      510         res  7,h                 ; ROM-Bit ruecksetzen
A015  226EA0    520         ld   (getmod),hl         ; in Programm eintragen
                530 ;
A018  2AD4BD    540         ld   hl,(#BDD4)          ; Indirektion TXT WRITE CHAR
A01B  2275A0    550         ld   (txtwrt),hl         ; in Programm eintragen
                560 ;
                570 ;       TXT WRITE CHAR verbiegen
                580 ;
A01E  2167A0    590         ld   hl,wrt0             ; Adresse Textausgabe im Programm
A021  22D4BD    600         ld   (#BDD4),hl          ; in Indirektion TXT WRITE CHAR eintragen
                610 ;
                620 ;       Controlcodetabelle verbiegen
                630 ;
A024  CDB1BB    640         call #BBB1               ; TXT GET CONTROLS
A027  E5        650         push hl                  ; Adresse Controlcodetabelle sichern
                660 ;
A028  111800    670         ld   de,24               ; Code fuer CHR$(24)
A02B  19        680         add  hl,de
A02C  19        690         add  hl,de
A02D  19        700         add  hl,de               ; Adresse Controlcode fuer CHR$(24)
A02E  23        710         inc  hl                  ; Sprungadresse fuer CHR$(24)
A02F  01A9A0    720         ld   bc,inv0             ; Einsprungadresse zum Invertieren
A032  71        730         ld   (hl),c
A033  23        740         inc  hl
A034  70        750         ld   (hl),b              ; als Adresse Controlcode CHR$(24) eintra
gen
                760 ;
A035  E1        770         pop  hl                  ; Adresse Controlcodetabelle wiederholen
A036  111B00    780         ld   de,27               ; Code fuer CHR$(27)
A039  19        790         add  hl,de
A03A  19        800         add  hl,de
A03B  19        810         add  hl,de               ; Adresse Controlcode fuer CHR$(27)
A03C  3601      820         ld   (hl),1              ; Anzahl Parameter auf eins setzen
A03E  23        830         inc  hl
A03F  01B8A0    840         ld   bc,funct0           ; Einsprungadresse Steuerfunktionen
A042  71        850         ld   (hl),c
A043  23        860         inc  hl
A044  70        870         ld   (hl),b              ; als Adresse Controlcode CHR$(27) eintra
gen
                880 ;
                890 ;       Ende Initialisierung
                900 ;
A045  C9        910         ret
                920 ;
                930 ;
                940 ;
                950 ;       Variablendefinitionen
                960 ;
                970 ;
                980 ;
A046  80        990 steuer: defb #80                 ; Steuerflag (bit 0-6 fuer Funktionen)
A047  00000000 1000 char:   defb 0,0,0,0,0,0,0,0     ; Ergaenzungsmatrix
A04F  00000000 1010 invert: defb 0,0,0,0,0,0,0,0     ; Invertierungsmatrix
A057  81422418 1020 xtab:   defb #81,#42,#24,#18,#18,#24,#42,#81; x - en
A05F  00010204 1030 sttab:  defb #00,#01,#02,#04,#08,#10,#20,#40; Bittabelle
               1040 ;
```

Listing 3. Der Quellcode für »Bildschirm de Luxe« (Fortsetzung)

Professionelle Grafikprogramme für Schneider CPC 6128 + Joyce

DIGITAL RESEARCH®

DR Draw

DR Draw: Macht aus Ihren Ideen ein Kunstwerk

Verwenden Sie DR Draw, um Organisations-Diagramme, Flußdiagramme, Logos, technische Zeichnungen, Schaubilder, Platinenentwürfe und jede nur erdenkliche Art von Linien- und Formgrafiken zu entwerfen. Und jeder Bestandteil Ihrer Zeichnung kann auf vielfältige Weise durch Farben und Schraffuren hervorgehoben werden.

Einfachste Bedienung

DR Draw verwendet leichtverständliche Menüs zur Steuerung seiner Funktionen und Erstellung einer Zeichnung. Sie können aus vorprogrammierten Figuren wie Kreisen, Quadern, Rechtecken, Kreisbögen, Polygonen und Linien auswählen oder Ihre eigenen Figuren entwerfen oder die bestehenden verändern. An beliebigen Stellen kann erläuternder Text in eine Zeichnung eingefügt werden.

Flexibilität bei der Gestaltung

Jeder Teil einer Zeichnung kann auf Tastendruck überarbeitet und verändert werden: Figuren können mit Farben oder Mustern gefüllt werden; sie können vergrößert oder verkleinert oder an eine neue Position verschoben oder kopiert werden. Ebenso können die Schriftarten, Größen, Farben und Positionen mit wenigen Tastendrücken geändert werden.

Ausgabe auf Papier, Transparentfolie oder Film

Was immer Sie erstellen, kann gespeichert oder zu Berichts- und Präsentationszwecken auf Papier, Transparentfolie oder Film geplottet oder gedruckt werden. DR Draw druckt Ihre Zeichnung exakt auf eine DIN-A4-Seite.

Hardwarevoraussetzungen

DR Draw läuft auf jedem Schneider CPC 6128 oder Joyce PCW 8256 mit einem oder zwei Diskettenlaufwerken. Die Grafiken können auf jedem Drucker oder Plotter ausgegeben werden, für den ein GSX-Treiber verfügbar ist. Dazu zählen Schneider-, Epson- und Shinwa-Drucker sowie der Plotter HP 7470A.

Die Fähigkeiten auf einen Blick

- Erstellung beliebiger Zeichnungen
- vorprogrammierte Figuren wie Kreise, Quader, Rechtecke, Kreisbögen, Polygone und Linien
- freie Wahl der Gestaltungselemente wie Farben, Muster und Schriftarten
- Vergrößerungen und Ausschnittdarstellungen
- Teile einer Zeichnung können kopiert, verschoben oder gelöscht werden
- Grafiken können gespeichert, geplottet oder gedruckt werden
- einfache Bedienung durch Menüauswahl

Best.-Nr. MS 613 **DM 199,–*** (sFr. 178,–/öS 1890,–*)

DR Graph

DR Graph: Präsentationsgrafiken mit professionellem Niveau

DR Graph ist ein interaktives Softwarepaket, mit dem Sie Ihren Mikrocomputer zur Erstellung von Geschäftsgrafiken und Text-Charts verwenden können. DR Graph macht es leicht, komplexe geschäftliche oder wissenschaftliche Daten in übersichtliche und aussagekräftige Grafiken zu verwandeln.

Einfachste Bedienung

Mit DR Graph können Sie die Grafik dem Computer schnell und leicht beschreiben. Zur Erstellung einer Grafik werden die gewünschten Optionen ganz einfach aus übersichtlichen Menüs ausgewählt. DR Graph kann von jedermann bedient werden, der mit einfachen Grundlagen der Mikrocomputerbedienung vertraut ist.

Flexibilität bei der Gestaltung

Zusätzlich zur vorhandenen Computerschrift stehen drei verschiedene Schriften für Titelzeilen, Legenden und Anmerkungen zur Verfügung. Auch bei der Gestaltung der Grafiken kann aus zahlreichen Linientypen, Linien- und Balkenbreiten und acht Schraffuren gewählt werden.

Ansehen, speichern und drucken

Mit DR Graph können Sie auf dem Bildschirm immer genau sehen, wie Sie Ihre Grafik gestalten. Anschließend können Sie sie drucken oder auf Diskette speichern, um sie später weiter zu bearbeiten.

Hardwarevoraussetzungen

DR Graph läuft auf jedem Schneider CPC 6128 oder Joyce PCW 8256 mit einem oder zwei Diskettenlaufwerken. Die Grafiken können auf jedem Drucker oder Plotter ausgegeben werden, für den ein GSX-Treiber verfügbar ist. Dazu zählen Schneider-, Epson- und Shinwa-Drucker sowie der Plotter HP 7470A.

Die Fähigkeiten auf einen Blick

- Linien-Grafiken, Histogramme, Torten-Grafiken, Stufen-Grafiken, Strich-Histogramme, Punkte-Grafiken und Text-Grafiken
- freie Wahl der Gestaltungselemente wie Beschriftungen, Titelzeilen, Legenden, Farben, Schriftarten und Ränder
- frei wählbare Skalierung
- variable Linien- und Balkenbreite
- Schnittstelle zu anderen Programmen
- beliebig positionierbare Anmerkungen
- Grafiken können gespeichert, geplottet oder gedruckt werden
- einfache Bedienung durch Menüauswahl

Best.-Nr. MS 614 **DM 199,–*** (sFr. 178,–/öS 1890,–*)

Jetzt neu!

Finanzbuchhaltung

Das Komplett-Paket für den Schneider CPC 6128 und Joyce. Erstellen von Kontenplänen, Umsatzsteuerauswertung und Einnahmen-/Überschußrechnung. Betriebswirtschaftliche Auswertungen wie Journalschreibung und Kostenstellenrechnung möglich.

Schneider CPC 6128 / 2 Laufwerke, Best.-Nr. MS 615 (3"-Diskette) DM 194,–* (sFr. 175,–)
Schneider Joyce PCW 8256 / 1 Laufwerk, Best.-Nr. MS 618 (3"-Disk.) DM 194,–* (sFr. 175,–)
Schneider Joyce PCW 8512 / 2 Laufwerke, Best.-Nr. MS 623 (3"-Disk.) DM 194,–* (sFr. 175,–)

Fakturierung

Ein dBASE-II-Anwenderprogramm, das folgende Möglichkeiten bietet: Angebotsschreibung und Rechnungsschreibung, Artikelverwaltung, Adreßverwaltung, Nachkalkulation. Der dokumentierte Quellcode wird für individuelle Programmanpassungen mitgeliefert.

Schneider CPC 6128/2 Laufwerke, Best.-Nr. MS 616 (3"-Disk.) DM 94,–* (sFr. 82,–/öS 940,–*)
Schneider Joyce PCW 8256/1 Laufw., Best.-Nr. MS 619 (3"-Disk.) DM 94,–* (sFr. 82,–/öS 940,–*)
Schneider Joyce PCW 8512/2 Laufw., Best.-Nr. MS 624 (3"-Disk.) DM 94,–* (sFr. 82,–/öS 940,–*)

* inkl. MwSt. Unverbindliche Preisempfehlung

Diese Markt&Technik-Softwareprodukte erhalten Sie in den Fachabteilungen der Kaufhäuser und in Computershops.



Unternehmensbereich Buchverlag
Hans-Pinsel-Straße 2, 8013 Haar bei München

Bestellungen im Ausland bitte an untenstehende Adressen.

Schweiz: Markt&Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel. 042/41 56 56

Österreich: Ueberreuter Media Handels- und Verlagsges. mbH, Alser Straße 24, A-1091 Wien, 0222/48 15 38-0

```
                    1050 ;
                    1060 ;
                    1070 ;
                    1080 ;          Ausgabe von Zeichen auf Bildschirm
                    1090 ;
                    1100 ;          ersetzt TXT WRITE CHAR
                    1110 ;
                    1120 ;
                    1130 ;
A067  E5            1140 wrt0:    push hl              ; Zeile / Spalte sichern
A068  4F            1150          ld   c,a             ; a sichern
                    1160 ;
A069  CD            1170          defb #CD
A06A  0000          1180 getmat: defw #0000            ; CALL TXT GET MATRIX
                    1190 ;
A06C  EB            1200          ex   de,hl           ; Adresse der Matrix dach de
                    1210 ;
A06D  CD            1220          defb #CD
A06E  0000          1230 getmod: defw #0000            ; CALL SCR GET MODE
A070  FE02          1240          cp   #02             ; ist Mode gleich 2 ?
                    1250 ;
A072  79            1260          ld   a,c             ; a wiederherstellen
A073  E1            1270          pop  hl              ; Zeile / Spalte wiederherstellen
A074  C2            1280          defb #C2
A075  0000          1290 txtwrt: defw #0000            ; JP NZ,TXT WRITE CHAR
                    1300 ;
A077  CD            1310          defb #CD
A078  0000          1320 chrpos: defw #0000            ; CALL SCR CHAR POSITION
                    1330 ;
A07A  FDE5          1340          push iy
A07C  FD2147A0      1350          ld   iy,char         ; Ergaenzungsmatrix
A080  EB            1360          ex   de,hl
                    1370 ;
                    1380 ;
                    1390 ;       Matrix ergaenzen und in den Bildschirmspeicher
                    1400 ;       uebertragen
                    1410 ;
                    1420 ;
A081  0608          1430          ld   b,8
A083  3A46A0        1440          ld   a,(steuer)
A086  4F            1450          ld   c,a             ; Steuerbyte nach c
A087  7E            1460 wrt10:   ld   a,(hl)          ; Zeichenmatrix nach a
                    1470 ;
A088  CB49          1480          bit  1,c             ; Doppeldruck / Fettschrift ein ?
A08A  2803          1490          jr   z,wrt13         ; nein ==>
A08C  CB3F          1500          srl  a               ; Matrixbyte nach rechts schieben
A08E  B6            1510          or   (hl)            ; mit Originalmatrix "OR"
                    1520 ;
A08F  CB51          1530 wrt13:   bit  2,c             ; Schmalschrift / Kursivschrift ein ?
A091  2803          1540          jr   z,wrt15         ; nein ==>
A093  CB27          1550          sla  a               ; Matrixbyte nach links schieben
A095  A6            1560          and  (hl)            ; mit Originalmatrix "AND"
                    1570 ;
A096  FDB600        1580 wrt15:   or   (iy+#00)        ; mit Ergaenzungsmatrix "OR"
A099  FDAE08        1590          xor  (iy+invert-char) ; invertieren
A09C  12            1600          ld   (de),a          ; Matrixbyte auf den Bildschirm bring
                    1610 ;
A09D  7A            1620          ld   a,d
A09E  C608          1630          add  a,#08
A0A0  57            1640          ld   d,a             ; eine Zeile tiefer
                    1650 ;
A0A1  23            1660          inc  hl              ; naechstes Matrixbyte
A0A2  FD23          1670          inc  iy              ; naechstes Ergaenzungsbyte
                    1680 ;
A0A4  10E1          1690          djnz wrt10           ; Matrix fertig ?
                    1700 ;
                    1710 ;
                    1720 ;
A0A6  FDE1          1730          pop  iy
A0A8  C9            1740          ret
                    1750 ;
                    1760 ;
                    1770 ;
                    1780 ;          Invertieren FLIP - FLOP
                    1790 ;
                    1800 ;
                    1810 ;
A0A9  214FA0        1820 inv0:    ld   hl,invert
A0AC  7E            1830          ld   a,(hl)          ; Invertierungsbyte nach a
A0AD  EEFF          1840          xor  #FF             ; "umdrehen"
A0AF  0608          1850          ld   b,8
A0B1  77            1860 inv3:    ld   (hl),a
A0B2  23            1870          inc  hl
A0B3  10FC          1880          djnz inv3
                    1890 ;
A0B5  C39CBB        1900          jp   #BB9C           ; TXT INVERSE
```

```
                        1910 ;
                        1920 ;
                        1930 ;
                        1940 ;          Behandlung von Steuerzeichen CHR$(27) + CHR$(x)
                        1950 ;
                        1960 ;
                        1970 ;
A0B8  2146A0            1980 funct0: ld    hl,steuer          ; Adresse Steuerbyte nach hl
A0BB  FE00              1990         cp    0                  ; Paramater a = 0  ?
A0BD  2003              2000         jr    nz,funct4          ; nein ==>
A0BF  CBBE              2010         res   7,(hl)             ; Highbit Steuerbyte loeschen
A0C1  C9                2020         ret
                        2030 ;
A0C2  CB7F              2040 funct4: bit   7,a                ; highbit im Parameter gesetzt ?
A0C4  2802              2050         jr    z,funct6           ; nein ==>
A0C6  CBFE              2060         set   7,(hl)             ; Highbit Steuerbyte setzen
                        2070 ;
A0C8  CB7E              2080 funct6: bit   7,(hl)             ; Highbit Steuerbyte gesetzt ?
A0CA  C8                2090         ret   z                  ; RETURN wenn nicht
                        2100 ;
A0CB  FE08              2110         cp    8                  ; Parameter > 7 ?
A0CD  D0                2120         ret   nc                 ; ja ==> RETURN
                        2130 ;
                        2140 ;
                        2150 ;     a-tes Bit des Steuerbyte umdrehen
                        2160 ;
                        2170 ;
A0CE  215FA0            2180         ld    hl,sttab           ; Adresse Bittabelle Steuerbyte
A0D1  1600              2190         ld    d,0
A0D3  5F                2200         ld    e,a
A0D4  19                2210         add   hl,de
A0D5  3A46A0            2220         ld    a,(steuer)         ; Steuerbyte nach a
A0D8  AE                2230         xor   (hl)               ; entsprechendes Bit umdrehen
A0D9  3246A0            2240         ld    (steuer),a         ; abspeichern
                        2250 ;
                        2260 ;
                        2270 ;     gemaess a springen
                        2280 ;
                        2290 ;
A0DC  79                2300         ld    a,c                ; a wiederherstellen
A0DD  E607              2310         and   7                  ; a modulo 7
A0DF  21ECA0            2320         ld    hl,funtab          ; Adresse Sprungtabelle nach hl
A0E2  1600              2330         ld    d,0
A0E4  5F                2340         ld    e,a                ; a nach de
A0E5  19                2350         add   hl,de
A0E6  19                2360         add   hl,de
A0E7  5E                2370 jphlin: ld    e,(hl)
A0E8  23                2380         inc   hl
A0E9  56                2390         ld    d,(hl)
A0EA  D5                2400 jpde:   push  de
A0EB  C9                2410         ret                      ; jp (de)
                        2420 ;
                        2430 ;
                        2440 ;     Sprungtabelle fuer Parameter
                        2450 ;
                        2460 ;
A0EC  05A1              2470 funtab: defw  funt20
A0EE  FCA0              2480         defw  funt10
A0F0  05A1              2490         defw  funt20
A0F2  06A1              2500         defw  funt30
A0F4  07A1              2510         defw  funt40
A0F6  0AA1              2520         defw  funt50
A0F8  13A1              2530         defw  funt60
A0FA  1CA1              2540         defw  funt70             ; Sprungadressen fuer Param 1-7
                        2550 ;
                        2560 ;
                        2570 ;     Param 1 : Unterstreichen ein/aus
                        2580 ;
                        2590 ;
A0FC  3A4EA0            2600 funt10: ld    a,(char+7)         ; Ergaenzungsmatrix, letztes Byte
A0FF  EEFF              2610         xor   #FF                ; umdrehen
A101  324EA0            2620         ld    (char+7),a         ; abspeichern
                        2630 ;
A104  C9                2640         ret
                        2650 ;
                        2660 ;
                        2670 ;     Param 2 : Doppeldruck / Fett ein/aus
                        2680 ;
                        2690 ;
A105  C9                2700 funt20: ret
                        2710 ;
                        2720 ;
                        2730 ;     Param 3 : Kursiv / Schmaldruck ein/aus
                        2740 ;
```

Listing 3. Der Quellcode für »Bildschirm de Luxe« (Fortsetzung)



```
                 2750 ;
A106  C9         2760 funt30: ret
                 2770 ;
                 2780 ;
                 2790 ;        Param 4 : Invertieren ein/aus
                 2800 ;
                 2810 ;
A107  C3A9A0     2820 funt40: jp    inv0               ; JUMP invertieren
                 2830 ;
                 2840 ;
                 2850 ;        Param 5 : durchstreichen ein/aus
                 2860 ;
                 2870 ;
A10A  3A4AA0     2880 funt50: ld    a,(char+3)         ; 4. Byte der Ergaenzungsmatrix
A10D  EEFF       2890         xor   #FF                ; umdrehen
A10F  324AA0     2900         ld    (char+3),a         ; abspeichern
                 2910 ;
A112  C9         2920         ret
                 2930 ;
                 2940 ;
                 2950 ;        Param 6 : ueberstreichen ein/aus
                 2960 ;
                 2970 ;
A113  3A47A0     2980 funt60: ld    a,(char)           ; 1. Byte der Ergaenzungsmatrix
A116  EEFF       2990         xor   #FF
A118  3247A0     3000         ld    (char),a           ; abspeichern
                 3010 ;
A11B  C9         3020         ret
                 3030 ;
                 3040 ;
                 3050 ;        Param 7 : durch - x - en  ein/aus
                 3060 ;
                 3070 ;
A11C  FD2147A0   3080 funt70: ld    iy,char            ; Adresse Ergaenzungstabelle
A120  0608       3090         ld    b,8
A122  FD7E00     3100 funt73: ld    a,(iy+#00)         ; Ergaenzungsbyte
A125  FDAE10     3110         xor   (iy+xtab-char)     ; Tabelle durch-x-en
A128  FD7700     3120         ld    (iy+#00),a         ; abspeichern
A12B  FD23       3130         inc   iy                 ; Zeiger erhoehen
A12D  10F3       3140         djnz  funt73
                 3150 ;
A12F  C9         3160         ret
```

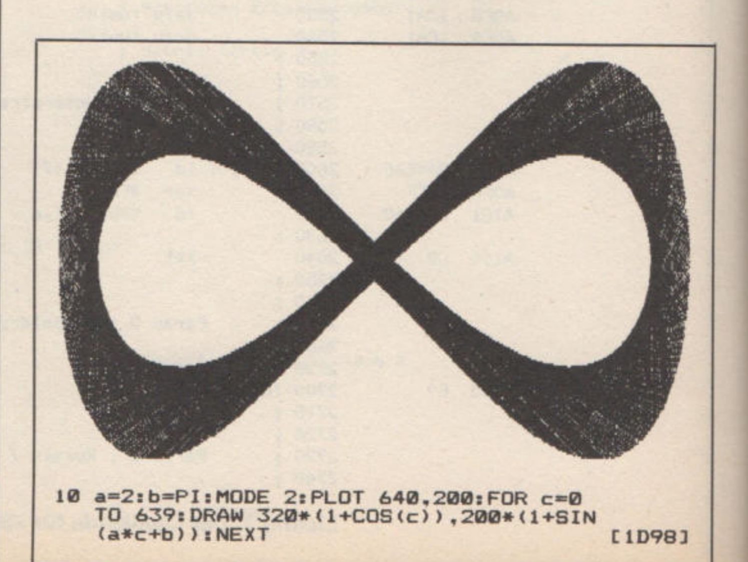
Listing 3. Der Quellcode »Bildschirm de Luxe« (Schluß)

```
10 DATA 2A,A6,BB,CB,BC,22,6A,A0,2A,1B,BC
   ,CB,BC,22,78,A0,2A,12,BC               [3C56]
20 DATA CB,BC,22,6E,A0,2A,D4,BD,22,75,A0
   ,21,67,A0,22,D4,BD,CD,B1               [A3FA]
30 DATA BB,E5,11,18,00,19,19,19,23,01,A9
   ,A0,71,23,70,E1,11,1B,00               [8798]
40 DATA 19,19,19,36,01,23,01,B8,A0,71,23
   ,70,C9,80,00,00,00,00,00               [050E]
50 DATA 00,00,00,00,00,00,00,00,00,00,00
   ,81,42,24,18,18,24,42,81               [C466]
60 DATA 00,01,02,04,08,10,20,40,E5,4F,CD
   ,00,00,EB,CD,00                        [D526]
70 DATA 00,FE,02,79,E1,C2,00,00,CD,00,00
   ,FD,E5,FD,21,47,A0,EB,06               [6384]
80 DATA 08,3A,46,A0,4F,7E,CB,49,28,03,CB
   ,3F,B6,CB,51,28,03,CB,27               [F0C4]
90 DATA A6,FD,B6,00,FD,AE,08,12,7A,C6,08
   ,57,23,FD,23,10,E1,FD,E1               [66FE]
100 DATA C9,21,4F,A0,7E,EE,FF,06,08,77,2
    3,10,FC,C3,9C,BB,21,46,A0             [543C]
110 DATA FE,00,20,03,CB,BE,C9,CB,7F,28,0
    2,CB,FE,CB,7E,C8,FE,08,D0             [4FFA]
120 DATA 21,5F,A0,16,00,5F,19,3A,46,A0,A
    E,32,46,A0,79,E6,07,21,EC             [82A4]
130 DATA A0,16,00,5F,19,19,5E,23,56,D5,C
    9,05,A1,FC,A0,05,A1,06,A1             [6B98]
140 DATA 07,A1,0A,A1,13,A1,1C,A1,3A,4E,A
    0,EE,FF,32,4E,A0,C9,C9,C9             [F16C]
150 DATA C3,A9,A0,3A,4A,A0,EE,FF,32,4A,A
    0,C9,3A,47,A0,EE,FF,32,47             [02AC]
160 DATA A0,C9,FD,21,47,A0,06,08,FD,7E,0
    0,FD,AE,10,FD,77,00,FD,23             [6E56]
170 DATA 10,F3,C9                         [AB40]
180 MEMORY &9FFF                          [A4C6]
190 FOR z=1 TO &130                       [638C]
200     READ a$                           [4BA6]
210     a%=VAL("&"+a$)                    [1448]
220     POKE HIMEM+z,a%                   [1954]
230 NEXT                                  [78E8]
240 CALL &A000                            [63D2]
250 END                                   [9C1C]
```

Listing 4. Der Basic-Lader für »Bildschirm de Luxe«

```
10 MODE 2                                        [DFF0]
20 FOR z=1 TO 7                                  [EE16]
30 PRINT"Dieses ";CHR$(27);CHR$(z);"W o
   r t";CHR$(27);CHR$(z);" ist hervorgeh
   oben"                                         [D4A6]
40 NEXT z                                        [7BBA]
```

Listing 5. So können Sie Texte hervorheben



```
10 a=2:b=PI:MODE 2:PLOT 640,200:FOR c=0
   TO 639:DRAW 320*(1+COS(c)),200*(1+SIN
   (a*c+b)):NEXT                                 [1D98]
```

Einfach und doch sicher

Oft bietet ein einziger Befehl einen guten Kopierschutz. Testen Sie doch einfach mit Hilfe der Ladegeschwindigkeit, ob das Original vorliegt.

Vor allem wer Programme verkaufen will, sucht beim Schneider nach einem wirkungsvollen Kopierschutz. Auf »SAVE "datei",P« kann man sich nicht verlassen. Einen erheblich besseren Schutz stellt die Variation der Ladegeschwindigkeit dar.

Dieser Kopierschutz funktioniert nur bei Programmen auf Kassette. Das Programm kann weder auf eine andere Kassette noch auf eine Diskette überspielt werden. Der Schutz beruht auf der Eigenschaft der Schneider-Computer, Programme auf dem Recorder mit unterschiedlicher Geschwindigkeit aufzuzeichnen. Mit SPEED WRITE – oder der Systemroutine CAS SET SPEED – kann die Geschwindigkeit in einem sehr weiten Bereich festgelegt werden.

Beim Laden erkennt der Computer die Aufzeichnungsgeschwindigkeit. Er speichert dafür einen Code. Dieser Code steht nach dem Laden im RAM und ist von dort aus zu lesen. Gängige Kopierprogramme besitzen zwar oft die Fähigkeit, Programme mit unterschiedlicher Baudrate aufzuzeichnen. Aber die Variationsbreite ist groß, und wenn man sein Programm standardmäßig mit einem »krummen« Wert speichert, ist es sehr unwahrscheinlich, daß das Kopierprogramm gerade diese Geschwindigkeit zum Speichern wählt. Den Geschwindigkeitswert kann man dann zum Test auf Kopie benutzen. Stimmt die festgestellte Ladegeschwindigkeit nicht mehr mit dem im Programm festgelegten Wert überein, bedeutet das, daß der Benutzer das Programm kopiert hat.

Besitzer des CPC 664 oder des CPC 6128 müssen statt B8CF hex in den Zeilen 10 und 20 die Adresse B1E7 hex abfragen. Um das vorgestellte Programm auszuprobieren, bitte unbedingt vorher ein Programm von der Kassette laden. Sie erhalten dann mit der Routine dessen Baudrate.

Wenn eine Abfrage nach der Geschwindigkeit gut in einem Maschinencode-Programm versteckt ist, bietet dieser einfache Trick einen sicheren Kopierschutz. Sie dürfen natürlich nicht gerade die Standardgeschwindigkeiten 1000 und 2000 Baud verwenden. Nehmen Sie lieber beispielsweise 1100 und 2100 Baud. Die »krummen Werte« sind auch von wirklichen Crackern oft nur durch Ausprobieren herauszufinden.

(Martin Kotulla/hg)

```
10 IF PEEK(&B8CF)=129 THEN PRINT "SPEED
   WRITE 0":END                         [F60A]
20 IF PEEK(&B8CF)=66 THEN PRINT "SPEED W
   RITE 1":END                          [85AE]
30 PRINT "WEDER 0 NOCH 1"               [F1A8]
```

Listing. Testen Sie die Ladegeschwindigkeit des Programms

Programm-Steckbrief	
Name:	Kopierschutz
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette

Farben-Geflimmer

Viele Kassetten-Turbolader bringen während des Ladevorgangs den Bildrand zum »Flimmern«. So gelingt auch Ihnen dieser Effekt...

Der Basic-Lader »Border« erzeugt den nötigen Maschinencode und enthält eine Demonstration der Effekte. Mit INPUT sind die Werte für Streifenbreite und Zahl der Farben einzugeben. Sie werden dem Maschinen-Programm in den Adressen A000 und A001 hex bereitgestellt.

(Dieter Braun/ja)

```
10 '  ***********************************
   *                                    [2F4C]
20 '  *          Border-Effekte
   *                                    [B7C4]
30 '  *
   *                                    [D2BC]
40 '  *  Erzeugt mehrfarbigen Rand und
   *                                    [4A76]
50 '  *     wartet auf Tastendruck
   *                                    [5C76]
60 '  *
   *                                    [2EC2]
70 '  *  Farbanzahl und Streifenbreite
   *                                    [E5FA]
80 '  *             angebbar
   *                                    [162A]
90 '  *
   *                                    [52C8]
100 ' *    Copyright by Dieter Braun
    *                                   [EA2A]
110 ' *            Maerz 1986
    *                                   [C248]
120 ' ***********************************
    *                                   [9C70]
130 GOSUB 230                           [7ED2]
140 INPUT "Streifenbreite{5 SPACE}(1-255
    ) ";w                               [EF42]
150 IF w<1 OR w>255 THEN 140            [B8FE]
160 INPUT "Anzahl der Farben{2 SPACE}(1-
    32){2 SPACE}";a                     [BF04]
170 IF a<1 OR a>32 THEN 160             [1540]
180 PRINT                               [6D8C]
190 POKE &A000,w                        [BF46]
200 POKE &A001,a                        [760C]
210 CALL &A010                          [7ECE]
220 RUN 140                             [17BC]
230 MEMORY &9FFF                        [9FBE]
240 DATA &3A,&00,&A0,&47,&3A,&01,&A0,&4F
                                        [2E88]
250 DATA &16,&00,&C5,&01,&00,&7F,&3E,&11
                                        [7358]
260 DATA &ED,&79,&AF,&3E,&40,&82,&ED,&79
                                        [812A]
270 DATA &C1,&C5,&10,&FE,&C1,&14,&79,&BA
                                        [58F6]
280 DATA &20,&E8,&CD,&09,&BB,&30,&D9,&C9
                                        [470E]
290 FOR n=40976 TO 41015                [46AA]
300 READ a:POKE n,a                     [AAA8]
310 NEXT n                              [3402]
320 RETURN                              [AC2A]
330 ' Gute Zahlenpaare:                 [8EAE]
340 '                                   [06BC]
350 ' 178/24  115/12  20/15  31/4   205/
    7                                   [FE70]
360 ' 49/15   46/13   1/24 255/16   23/2
    7                                   [2268]
```

Listing. Bunte Spielereien

Programm-Steckbrief	
Name:	Border
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

Spion aus dem Stack

Programme, die keinen oder kaum RAM-Speicherplatz belegen, gibt es eigentlich nur auf ROM-Basis. »RSX-Dump« aber schafft es, sich im Arbeitsspeicher zu »verstecken«.

Jeder Programmierer, der sich mit Maschinensprache auseinandersetzt, besitzt sicher das eine oder andere Programm, mit dessen Hilfe er sich Inhalte von Speicherbereichen anzeigen läßt. Wenn nun jedoch das zu untersuchende Programm zufällig im gleichen Adreßbereich des Arbeitsspeichers liegt oder diesen gar komplett ausfüllt, ist guter Rat teuer. Nur Besitzer entsprechender ROM-Software (beispielsweise Arnors »Maxam«, Vortex-Speichererweiterung oder VDOS 2) sind aus dem Schneider, denn diese belegen in kaum nennenswertem Umfang RAM-Speicher. Daß man auch mit »normaler« Software, die sich im Arbeitsspeicher befindet, solche Probleme vermeiden kann, beweist »RSX-Dump«. Es belegt die Adressen AF00 bis B070 hex. Diesen Bereich nutzt normalerweise das Locomotive-Basic als Software-Stack. Ab B08A hex (B06E hex beim CPC 664 und 6128) legt der Interpreter dort Werte zwischendurch ab. Dieser Stack »wächst« in Richtung niedrigerer Adressen hinunter bis nach AE8B hex (AE6F hex). Somit ist klar, daß »RSX-Dump« nicht mit Basic-Programmen verwendet werden sollte. Nur wenn es sich um kleinere Programme handelt, ist ein Überschreiben der RSX-Erweiterung durch den Stack vermeidbar. Da ein solches Tool aber ohnehin meist als eigenständiges Hilfsmittel eingesetzt wird, ergeben sich in der Praxis keinerlei Probleme. Schon der Name »RSX-Dump« sagt, wie die Befehlserweiterung in das Betriebssystem der Schneider-Computer eingebunden ist. Der Befehl DUMP beginnt also mit dem senkrechten Balken, der bei gleichzeitigem Druck der Tasten »SHIFT« und »@« auf dem Bildschirm entsteht. Tippen Sie zunächst das Listing des Basic-Laders ab und speichern Sie es zur Sicherheit. Beim Starten mit »RUN« erzeugt der Lader automatisch das Maschinencode-Programm »DUMP.BIN«, das Sie dann zum späteren Gebrauch benötigen. Sie laden es durch »LOAD "DUMP.BIN", &AF00«. Nach »CALL &AF00« steht Ihnen der neue Befehl zur ständigen Verfügung. Drei Parameter sind beim Aufruf zu übergeben: Start- und Endadresse des Speicherbereichs sowie der Ausgabekanal.

»IDUMP,&100,&200,0« listet den Inhalt der 256 Byte zwischen 100 hex und 200 hex auf dem Bildschirm auf. »IDUMP,1024,42619,8« gibt den Bereich von Adresse 1024 dez bis zum Beginn des Disketten-Controller-RAMs über den Drucker aus. Wie von den meisten Monitor- und Debugger-Programmen gewohnt, steht rechts neben der Angabe der Bytes die entsprechende ASCII-Interpretation zur schnellen Erkennung von Textfeldern.

(Raimund Schuldhaus/ja)

Programm-Steckbrief	
Name:	RSX-Dump
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette
Besonderes:	belegt Basic-Stack

```
A000: 01 09 AF 21 13 AF C3 D1 BC 0E AF C3 17 AF 44 55  ../!./CQ<./C./DU
A010: 4D D0 00 00 00 00 00 FE 03 DA 69 B0 C2 5D B0 DD  MP.....~.Zi0B]0]
A020: 7E 01 B7 C2 61 B0 DD 7E 00 FE 09 D2 61 B0 32 56  ~.7Ba0]~.~.Ra02V
A030: B0 FE 08 20 07 3E 10 32 57 B0 18 11 CD 11 BC B7  0~. .>.2W0..M.<7
A040: CA 65 B0 47 3E 04 CB 27 10 FC 32 57 B0 DD 5E 02  Je0G>.K'.|2W0]^.
A050: DD 56 03 DD 6E 04 DD 66 05 E5 B7 ED 52 E1 DA 64  ]V.]n.]f.e7mRaZd
A060: AF C2 61 B0 ED 53 59 B0 54 5D 3E 0D CD 1A B0 AF  /Ba0mSY0T]>.M.0/
A070: 32 58 B0 3A 57 B0 47 7A CD F9 AF CD 1A B0 79 CD  2X0:W0GzMy/M.0yM
A080: 1A B0 7B CD F9 AF CD 1A B0 79 CD 1A B0 3E 3A CD  .0{My/M.0yM.0>:M
A090: 1A B0 3E 20 CD 1A B0 ED 53 5B B0 1A CD F9 AF CD  .0> M.0mS[0.My/M
A0A0: 1A B0 79 CD 1A B0 3E 20 CD 1A B0 13 10 ED 3E 20  .0yM.0> M.0..m> 
A0B0: CD 1A B0 ED 5B 5B B0 3A 57 B0 47 1A FE 20 30 02  M.0m[[0:W0G.~ 0.
A0C0: 3E 2E FE 7F 20 02 3E 2E CD 1A B0 13 10 ED 3A 56  >.~. .>.M.0..m:V
A0D0: B0 FE 08 28 10 3A 58 B0 FE 18 20 05 CD 18 BB 3E  0~.(.:X0~. .M.;>
A0E0: 00 3C 32 58 B0 3E 0D CD 1A B0 3E 0A CD 1A B0 2A  .<2X0>.M.0>.M.0*
A0F0: 59 B0 B7 ED 52 D2 73 AF C9 F5 CB 3F CB 3F CB 3F  Y07mRRs/IuK?K?K?
A100: CB 3F FE 0A 38 02 C6 07 C6 30 4F F1 E6 0F FE 0A  K?~.8.F.F0Oqf.~.
A110: 38 02 C6 07 C6 30 67 79 4C C9 F5 C5 3E 42 CD 1E  8.F.F0gyLIuE>BM.
A120: BB C1 28 03 F1 E1 C9 3A 56 B0 FE 08 28 05 F1 CD  ;A(.qaI:V0~.(.qM
A130: 5A BB C9 F1 F5 FE 80 38 13 3E 1B CD 4D B0 3E 3D  Z;Iqu~8.>.MM0>=
```

Eine typische Ausgabe auf Drucker

```
100 '*********************************** [CED8]
110 '*     Hex-Dump (c) 12.1985 by     * [AC18]
120 '*        Raimund Schuldhaus       * [89E4]
130 '* Marburger Str.26, 6300 Giessen * [B610]
140 '*********************************** [CAE0]
150 '                                    [E1BA]
160 DATA 01,09,AF,21,13,AF,C3,D1,BC,0E,A
    F,C3,17,AF,44,55,-1739               [63E8]
170 DATA 4D,D0,00,00,00,00,00,FE,03,DA,6
    9,B0,C2,5D,B0,DD,-1725               [B06C]
180 DATA 7E,01,B7,C2,61,B0,DD,7E,00,FE,0
    9,D2,61,B0,32,56,-2006               [7C86]
190 DATA B0,FE,08,20,07,3E,10,32,57,B0,1
    8,11,CD,11,BC,B7,-1502               [DC4A]
200 DATA CA,65,B0,47,3E,04,CB,27,10,FC,3
    2,57,B0,DD,5E,02,-1756               [63A6]
210 DATA DD,56,03,DD,6E,04,DD,66,05,E5,B
    7,ED,52,E1,DA,64,-2247               [C90E]
220 DATA AF,C2,61,B0,ED,53,59,B0,54,5D,3
    E,0D,CD,1A,B0,AF,-2061               [7814]
230 DATA 32,58,B0,3A,57,B0,47,7A,CD,F9,A
    F,CD,1A,B0,79,CD,-2190               [EA20]
240 DATA 1A,B0,7B,CD,F9,AF,CD,1A,B0,79,C
    D,1A,B0,3E,3A,CD,-2214               [2392]
250 DATA 1A,B0,3E,20,CD,1A,B0,ED,53,5B,B
    0,1A,CD,F9,AF,CD,-2150               [C75E]
260 DATA 1A,B0,79,CD,1A,B0,3E,20,CD,1A,B
    0,13,10,ED,3E,20,-1597               [BAC2]
270 DATA CD,1A,B0,ED,5B,5B,B0,3A,57,B0,4
    7,1A,FE,20,30,02,-1756               [A3D0]
280 DATA 3E,2E,FE,7F,20,02,3E,2E,CD,1A,B
    0,13,10,ED,3A,56,-1454               [44E2]
290 DATA B0,FE,08,28,10,3A,58,B0,FE,18,2
    0,05,CD,18,BB,3E,-1609               [05B8]
300 DATA FF,3C,32,58,B0,3E,0D,CD,1A,B0,3
    E,0A,CD,1A,B0,2A,-1632               [2014]
310 DATA 59,B0,B7,ED,52,D2,73,AF,C9,F5,C
    B,3F,CB,3F,CB,3F,-2511               [B38A]
320 DATA CB,3F,FE,0A,38,02,C6,07,C6,30,4
    F,F1,E6,0F,FE,0A,-1868               [1028]
330 DATA 38,02,C6,07,C6,30,67,79,4C,C9,F
    5,C5,3E,42,CD,1E,-1815               [CFA6]
340 DATA BB,C1,28,03,F1,E1,C9,3A,56,B0,F
    E,08,28,05,F1,CD,-2163               [F8D2]
350 DATA 5A,BB,C9,F1,F5,FE,80,38,13,3E,1
    B,CD,4D,B0,3E,3D,-2091               [AF40]
360 DATA CD,4D,B0,F1,CD,4D,B0,AF,CD,4D,B
    0,C9,F1,CD,2E,BD,-2672               [01F2]
370 DATA 38,FB,CD,31,BD,C9,00,00,00,00,0
    0,00,00,1E,02,18,-1007               [94D4]
380 DATA 0A,1E,05,18,06,1E,06,18,02,1E,1
    6,CD,00,B9,C3,94,-922                [BCC6]
390 DATA CA,-202                         [BCCE]
400 MEMORY &9FFF                         [9FBC]
410 adr=&A000                            [D440]
420 FOR i%=10 TO 33                      [4B64]
430   READ op$:IF LEFT$(op$,1)="-"THEN 4
    60                                   [5B6C]
440   op%=VAL("&"+op$):POKE adr,op%      [78EA]
450   adr=adr+1:psum%=psum%+op%:GOTO 430
                                         [C748]
460   IF psum%+VAL(op$)=0 THEN 480       [F1DA]
470   PRINT"{2 SPACE}DATA-Fehler in Zeil
    e";i%*10:END                         [C63C]
480 psum%=0:NEXT                         [F218]
490 SAVE"dump.bin",b,&A000,369           [5898]
```

Listing. Hexdump aus dem Stack

Blitzschnell im Programm

Schnell genug kann es eigentlich nie gehen. Deshalb beschleunigen Sie doch einfach Ihr 3-Zoll-Laufwerk mit dem Programm »Turbo-Disk«.

Die CPC-Besitzer sind ja eigentlich noch ganz gut dran, wenn man die Ladezeiten mit Laufwerken anderer Computer vergleicht. Und doch: Warum nicht Zeit sparen, wenn es leicht möglich ist? Der Basic-Lader erzeugt ein kleines Maschinencode-Programm, das Diskettenzugriffe um zirka 20 Prozent beschleunigt. Dieser Effekt hält so lange an, bis Sie Ihren Computer ausschalten oder mit »CTRL-SHIFT-ESC« einen Reset auslösen. Im CP/M-Betrieb benötigen Sie noch nicht mal ein eigenes Programm für den Geschwindigkeitsgewinn. Wenn Sie das Betriebssystem erst einmal mit »SETUP« manipuliert haben, wird der »Turbo-Lader« automatisch bei jedem CP/M-Start aktiviert. Die genaue Vorgehensweise Schritt für Schritt:

Programm-Steckbrief	
Name:	Turbo-Disk
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Diskette
Besonderes:	arbeitet nur auf 3-Zoll-Laufwerken

»Setup«, ENTER, Y, Y, Y, Y, Y, Y, Y, Y, Y, »N40«, ENTER, »YN200«, ENTER, »YN10«, ENTER, Y, Y, Y, Y, Y. Natürlich läßt sich dieses verbesserte CP/M mit »BOOTGEN« auf jede Arbeitsdiskette übertragen.

(Sebastian Scala/ja)

```
10  REM  ****************************     [52BA]
20  REM  *        TURBO DISK        *     [3CE2]
30  REM  *      (c) 1986 by         *     [0844]
40  REM  *     Sebastian  Scala     *     [2E34]
50  REM  *     Am Silberberg 20     *     [695C]
60  REM  *     A - 8042 Graz        *     [6DDC]
70  REM  *   Tel: (0316) 40-16-15   *     [9448]
80  REM  ****************************     [CAC8]
90  REM                                   [9E1A]
100 REM                                   [5E2A]
110 FOR a=0 TO 18                         [2DA6]
120 READ a$                               [3CE8]
130 POKE a+&A000,VAL("&"+a$)              [E900]
140 NEXT                                  [69E8]
150 CALL &A000                            [A0D2]
160 MODE 2:PEN 1:PAPER 0:INK 1,24:INK 0,
    0:BORDER 9                            [2B6C]
170 PRINT"TURBO DISK ist nun aktiviert,"
    :
         PRINT"ihre Floppy ist jetzt
    ca. um 1/5 schneller als vorher !":P
    RINT                                  [382A]
180 NEW                                   [7A46]
190 DATA 21,0A,A0,DF,07,A0,C9,0D,C6,07,2
    3,00,C8,00,01,01,0A,00,03             [3332]
```

Listing. Schnell, schneller, Turbo



Selbstuntersuchung

Maschinencode-Routinen sind das Salz in der Suppe der selbstgeschriebenen Programme. Will man diese aber willkürlich irgendwo im Speicher ablegen, dann treten einige Probleme auf.

Besonders schnelle Programme, beziehungsweise Unterprogramme, entstehen, wenn man sie in Maschinencode schreibt. Allerdings erfordert solch eine maschinennahe Sprache höchste Sorgfalt bei der Entwicklung. Während man in Basic Unterprogramme mit einem Befehl GOTO <Zeilennummer> aufrufen darf, muß man bei Maschinencode-Routinen die genaue Adresse im Speicher kennen. Sind mehrere Programme geladen, so kann es passieren, daß zwei an der gleichen Adresse stehen.

Gute Maschinencode-Routinen sind deshalb immer so geschrieben, daß sie frei verschiebbar sind. Das erreicht man, indem man absolute Adressen vermeidet. Zum Starten braucht man aber noch die Startadresse. Wie man die aus dem Computer »herauskitzelt«, das lesen Sie hier.

Ein Weg fällt den meisten Z80-Programmierern sofort ein. Man ruft ein Unterprogramm auf, das die Rücksprungadresse vom Stack holt und in ein Register überträgt. Listing 1 (Basic-Lader in Listing 2) zeigt solch eine Routine. Nach dem Aufruf der Routine liegt die Startadresse im Speicher an der Adresse 0160 und 0161 hex und kann dort mit »PRINT(PEEK (&0160)+PEEK (&0161)*256)« abgefragt werden. Diese Lösung hat aber einen schwerwiegenden Nachteil. Sie verlangt eine absolute Unterprogrammadresse, was wir ja gerade vermeiden wollten. Folglich wandert sie in die Abteilung P – sprich Papierkorb.

Eine bessere und auch kürzere Routine bietet Listing 3 (Basic-Lader Listing 4) an. Sie macht sich die Eigenschaft der Schneider-Computer zunutze, im Betriebssystem-RAM einen Zeiger auf ein CALL-Unterprogramm oder eine RSX-Routine zu speichern. Beim CPC 464 liegt dieser Zeiger bei AE72 hex, bei den Nachfolgemodellen 664 und 6128 aber an der Adresse AE55 hex. Und das ist der Nachteil. Denn man muß immer die auszulesende Adresse für den Zielcomputer einsetzen. Die Listings beziehen sich auf den CPC 464. Die Änderungen für die beiden Nachfolgermodelle stehen in Listing 5.

Erst die dritte Lösung entspricht unseren Anforderungen. Sie ist besonders universell, da sie weder absolute Sprünge innerhalb des Programms noch systemspezifische Adressen benutzt. Der Trick beruht auf den ausgeprägten Interruptfähigkeiten der Schneider-Geräte. Die beiden Maschinencode-Befehle »EI« lassen Interrupts zu und »HALT« stoppt alle Prozessoraktivitäten, bis der nächste Interrupt auftritt. Dieser läßt nicht lange auf sich warten, und hinterläßt auf dem Stack »als Geschenk an das Programm« die Rückkehradresse, die gleichzeitig die Programmadresse ist. Allerdings muß der Stackpointer SP mit einem zweifachen »DEC SP« manipuliert werden. Dann kann man aber die Adresse aus dem Register HL übernehmen. Dieses Verfahren beschränkt sich übrigens nicht nur auf den Schneider CPC, sondern läßt sich auf viele andere Z80-Computer übertragen. Listing 6 zeigt das Assembler-Programm und Listing 7 den zugehörigen Basic-Lader.

(Martin Kotulla/hg)

Programm-Steckbrief	
Name:	Call
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette, Diskette

```
              ORG     &A000

START         CALL    CALC_ADDRESS     ; Startadresse nach HL holen
              LD      (&0160),HL       ; In &0160 abspeichern
              RET                      ; Rücksprung nach Basic

CALC_ADDRESS  POP     HL               ; Rücksprungadresse nach HL
              PUSH    HL               ; Rücksprung wiederherstellen
              DEC     HL               ; Um drei Bytes die
              DEC     HL               ; Startadresse in HL
              DEC     HL               ; vermindern
              RET                      ; Rücksprung zum Hauptprogramm
              END                      ; ---------------------------------
```

Listing 1. Häufig bekannt, aber nicht optimal, ist die Lösung 1

```
10 DATA CD,07,A0,22,60,01,C9                  [B892]
20 DATA E1,E5,2B,2B,2B,C9                     [F2C0]
30 MEMORY &9FFF:FOR i=&A000 TO &A00C          [ADF0]
40 READ a$:POKE i,VAL("&"+a$):NEXT i          [1566]
50 CALL &A000                                 [9070]
60 PRINT "Die Startadresse liegt bei";        [5EBC]
70 PRINT PEEK(&160)+PEEK(&161)*256            [CD1A]
```

◀ Listing 2. Wer sich mit dem Assembler-Programm nicht anfreunden will, findet hier den Basic-Lader für Listing 1

```
              ORG     &A000

START         LD      HL,(&AE72)       ; Startadresse nach HL holen
              LD      (&0160),HL       ; In &0160 abspeichern
              RET                      ; Rücksprung nach Basic
              END                      ; ---------------------------------
```

Listing 3. Solange man nur einen Schneider-Computer benutzt, ist diese Lösung optimal

```
10 DATA 2A,72,AE,22,60,01,C9                    [FC98]
20 INPUT "Welche Startadresse? ",s               [7CC2]
30 MEMORY s-1:FOR i=s TO s+6                     [9CBC]
40 READ a$:POKE i,VAL("&"+a$):NEXT i             [1566]
50 CALL s                                        [7868]
60 PRINT "Die Startadresse liegt bei";           [5EBC]
70 PRINT PEEK(&160)+PEEK(&161)*256               [CD1A]
```

Listing 4. Der Basic-Lader für Listing 3

```
START    LD     HL,(&AE55)

10 DATA 2A,55,AE,22,60,01,C9
```

Listing 5. Für den CPC 664 und 6128 müssen Sie im Assembler-Programm die zweite Zeile und im Basic-Lader Zeile 10 ändern

```
         ORG    &A000

START    EI                ; Interrupts zulassen
         HALT              ; Warten auf Interrupt
         DEC    SP         ; Stackpointer um einen
         DEC    SP         ; Eintrag nach unten schieben
         POP    HL         ; Adresse holen
         DEC    HL         ; Berichtigen, EI-Code abziehen
         DEC    HL         ; Berichtigen, HALT abziehen
         LD     (&160),HL  ; Adresse speichern
         RET               ; Rücksprung nach Basic
```

Listing 6. Dieses Programm entspricht unseren Anforderungen

```
10 DATA FB,76,3B,3B,E1,2B,2B,22,60,01,C9
                                                 [62AE]
20 INPUT "Welche Startadresse? ",s               [7CC2]
30 MEMORY s-1:FOR i=s TO s+10                    [B012]
40 READ a$:POKE i,VAL("&"+a$):NEXT i             [1566]
50 CALL s                                        [7868]
60 PRINT "Die Startadresse liegt bei";           [5EBC]
70 PRINT PEEK(&160)+PEEK(&161)*256               [CD1A]
```

Listing 7. Frei verschiebbar, und die gesuchte Adresse steht in 160 und 161 hex

Tolle Video-Tricks

OUT heißt der Basic-Befehl, der den Video-Chip zu ungeahnten Fähigkeiten antreibt.

Der Basic-Befehl OUT gehört wie bei vielen Computern auch beim Schneider zu den nur sehr selten benutzten Kommandos. Der Grund liegt darin, daß fast niemand weiß, was man damit Sinnvolles anstellen kann. Und dabei gibt es viele reizvolle Möglichkeiten.

Auch den Video-Chip im Schneider, ein Motorola 6845, steuert der Computer über OUT-Befehle. Er besitzt eine Reihe von Registern, die Werte aufnehmen können, die wiederum die Arbeit manipulieren.

Die Programmierung des Chips ist somit auch von Basic aus möglich. Um einem Register einen Wert zu übermitteln, muß über die Portadresse BC00 hex die Registernummer ausgegeben werden und unmittelbar danach über BD00 hex der Wert. Wollen wir beispielsweise in das Register 5 den Wert 20 eintragen, sind folgende Befehle notwendig:

```
OUT &BC00,5:OUT &BD00,20
```

Das Register 2 (Horizontal Synchro Position) bestimmt, wo genau auf dem Monitor der Inhalt des Video-RAMs angezeigt wird.

(Martin Kotulla/hg)

Programm-Steckbrief	
Name:	OUT6845
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette, Diskette

```
100  ' Steuerung des Video-Chips                  [0DA2]
110  ' Motorola-6845 im Schneider-CPC             [4AF0]
120  ' ueber OUT-Befehle                          [09FA]
130  ' (c) Martin Kotulla                         [C46C]
140  ' ********************************           [D4CC]
150  '                                            [E1BA]
160 MODE 1                                        [BE5A]
170 FOR i=1 TO 25                                 [2EC0]
180 PRINT TAB(5);"DEMONSTRATION FUER OUT
    -BEFEHLE"                                     [A356]
190 NEXT i                                        [3C04]
200 ON BREAK GOSUB 270                            [455A]
210 ' Hauptprogramm *****************             [353E]
220 BORDER 1:INK 0,26                             [3886]
230 FOR i=0 TO 47:CALL &BD19:OUT &BC00,2
    :OUT &BD00,i:NEXT i                           [C438]
240 FOR i=47 TO 0 STEP -1:CALL &BD19:OUT
    &BC00,2:OUT &BD00,i:NEXT i                    [F1EE]
250 GOTO 230                                      [D64A]
260 ' Break-Routine ******************            [576A]
270 OUT &BC00,2:OUT &BD00,46                      [E85C]
280 OUT &BC00,7:OUT &BD00,30                      [1E5A]
290 END                                           [0824]
{CTRL Z}{CTRL J}130 ' (c) Martin Kotulla
                                                  [A7B4]
140 '                                             [02B8]

100  ' Steuerung des Video-Chips                  [0DA2]
110  ' Motorola-6845 im Schneider-CPC             [4AF0]
120  ' ueber OUT-Befehle                          [09FA]
130  ' (c) Martin Kotulla                         [C46C]
140  ' ********************************           [D4CC]
150  '                                            [E1BA]
160 MODE 1                                        [BE5A]
170 FOR i=1 TO 25                                 [2EC0]
180 PRINT TAB(5);"DEMONSTRATION FUER OUT
    -BEFEHLE"                                     [A356]
190 NEXT i                                        [3C04]
200 ON BREAK GOSUB 270                            [455A]
210 ' Hauptprogramm *****************             [353E]
220 BORDER 1:INK 0,26                             [3886]
230 FOR i=0 TO 47:CALL &BD19:OUT &BC00,2
    :OUT &BD00,i:NEXT i                           [C438]
240 FOR i=47 TO 0 STEP -1:CALL &BD19:OUT
    &BC00,2:OUT &BD00,i:NEXT i                    [F1EE]
250 GOTO 230                                      [D64A]
260 ' Break-Routine ******************            [576A]
270 OUT &BC00,2:OUT &BD00,46                      [E85C]
280 OUT &BC00,7:OUT &BD00,30                      [1E5A]
290 END                                           [0824]

                                                  [A7B4]
```

Listing. Tolle Effekte mit dem OUT-Befehl

Das »CAT-Syndrom«

Haben auch Sie Probleme mit der Kombination 3-Zoll-Laufwerk und Vortex-Speichererweiterung? Ihnen kann geholfen werden.

Wer ein 3-Zoll-Laufwerk besitzt und freudenstrahlend seine neu erworbene Speichererweiterung in Betrieb setzt, erlebt oft eine herbe Enttäuschung: Beim Befehl CAT produziert sein Laufwerk alles mögliche, nur kein sinnvolles Directory! Natürlich schlußfolgert er, daß der Fehler nur bei der Erweiterung zu suchen ist; vor dem Einbau hat schließlich noch alles einwandfrei funktioniert...

Und doch ist das nur die halbe Wahrheit. Durch die zusätzliche Elektronik ist zwar tatsächlich das System überlastet; der Hauptschuldige ist jedoch der Computer – oder vielmehr seine Hardware-Entwickler. Derartige Probleme treten nämlich prinzipiell bei Einsatz jeglicher zusätzlicher Peripherie ein. Überspitzt ausgedrückt sind die CPCs so konstruiert, daß sie keinen Anschluß weiteres Zubehör verkraften. Die Erklärung dafür ist einfach: Der Systembus ist bis hart an die Grenzen der Belastbarkeit ausgereizt. Und wenn dann die geringste Kleinigkeit dazukommt, führt das fast immer zum Zusammenbruch des Systems. Aufgrund von Streuungen in der Serie tritt dieser Effekt aber nicht immer auf. Eine ausweglose Situation? Mitnichten! Wenn Sie Zugriff auf einen EPROM-Brenner haben, ist Ihnen zu helfen: Kopieren Sie den Inhalt des Controller-ROMs in ein EPROM, denn es verfügt über eine höhere Treiberleistung als das ROM. (ja)

Gut gerüstet

Viele Benutzer kaufen sich aus preislichen Gründen die kleinste Stufe der Speichererweiterung, um sie dann selbst aufzurüsten. Doch ganz ohne Tücken ist das nicht.

Die Aufstockung der Speicherkapazität geschieht in der Praxis meist durch Bestücken der zweiten RAM-Bank. In diesem Falle müssen Sie auf den weiteren freien Sockel einen Logikbaustein setzen. Er dient der Dekodierung des zusätzlichen RAM-Bereichs. Dabei sollten Sie darauf achten, daß es sich um einen HCT-Typ handelt (74HCT378). Diese benötigen extrem wenig Strom und sind sehr schnell. Mit gewöhnlichen LS-Typen kann es zu Komplikationen kommen. Als RAM-Bausteine wählen Sie solche mit Zugriffszeiten von 120 Nanosekunden. Wer im Betriebssystem der Karte Änderungen vornehmen will, muß das ROM gegen ein gepatchtes EPROM tauschen. Einer einwandfreien Funktion dienen hier Typen mit 250 Nanosekunden Zugriffszeit.

Nur in Ausnahmefällen hat wohl ein Benutzer Interesse daran, die Platine öfters ein- und auszubauen. Um die Anschlußbeinchen des Gate-Arrays zu schonen, ist dann empfehlenswert, es mit einem eigenen Sockel auszustatten.

(Udo Reetz/ja)

Wordstar in der RAM-Disk

Wordstar und eine Speichererweiterung von Vortex. So wird Textverarbeitung wirklich komfortabel.

Wer Wordstar auf dem Schneider CPC unter CP/M 2.2 wirklich sinnvoll benutzen will, kommt um die Anschaffung einer Speichererweiterung meist nicht herum. Die weit verbreitete Vortex-Karte bietet unter CP/M eine RAM-Disk als Laufwerk C mit einer Kapazität von über 400 KByte (bei voller Ausbaustufe auf 512 KByte). Doch bei jedem Systemstart beispielsweise den kompletten Wordstar und den zu bearbeitenden Text in die Speichererweiterung zu übertragen, ist unnötige Arbeit. Das Programm SUBMIT.COM macht das einfacher. Kopieren Sie es dazu auf Ihre Wordstar-Diskette. Dann geben Sie die Submit-Datei aus Listing 1 als NEW.SUB mit dem Wordstar-Editor (N-Modus) ein. Listing 2 erhält den Namen ED.SUB.

Wenn Sie die RAM-Disk nicht gleich beim Start von CP/M mitformatieren, dann benötigen Sie noch das Programm RAMDISK.COM von Vortex auf Ihrer Wordstar-Diskette. Um einen neuen Text zu bearbeiten, laden Sie CP/M und tippen folgendes:

```
A>RAMDISK
A>SUBMIT NEW datei
```

Das Programm kopiert alle Wordstar-Teile und PIP.COM in die RAM-Disk und startet die Textverarbeitung. Danach können Sie Ihren Text bearbeiten. Drücken Sie CTRL-KX, wird SUBMIT wieder aktiv und überträgt die Datei auf die Originaldiskette im Laufwerk A zurück. Dort ist sie dauerhaft gespeichert. Beachten Sie aber, daß Sie beim Dateinamen »datei« keine Laufwerkbezeichnung angeben dürfen.

Ähnlich funktioniert ED.SUB (Listing 2). Es dient zur Bearbeitung einer bereits bestehenden Datei im Laufwerk A. Denn »ED« steht für »Editieren«.

```
A>RAMDISK
A>SUBMIT ED datei
```

Den Befehl »RAMDISK« können Sie natürlich auch in die Submit-Datei aufnehmen. Sie verlieren dann aber bei jedem Aufruf der Submit-Datei den vorherigen Inhalt der RAM-Disk. Wer die RAM-Disk beim Aufruf von CP/M mitformatiert, muß die Anweisung RAMDISK natürlich weglassen.

(Martin Kotulla/hg)

```
A:PIP C:=A:PIP.COM
C:PIP C:=A:WS.COM
C:PIP C:=A:WSMSGS.OVR
C:PIP C:=A:WSOVLY1.OVR
C:
WS $1
PIP A:=C:$1
A:
```

Listing 1. Mit dieser Datei wird Wordstar komfortabel

```
A:PIP C:=A:PIP.COM
C:PIP C:=A:WS.COM
C:PIP C:=A:WSMSGS.OVR
C:PIP C:=A:WSOVLY1.OVR
C:PIP C:=A:$1
C:
WS $1
PIP A:=C:$1
A:
```

Listing 2. Auch bereits bestehende Dateien können bearbeitet werden

Kopieren in die RAM-Floppy

Kopieren in die RAM-Floppy geht auch mit »Filecopy«. Vorausgesetzt man baut die Routine um.

Jeder, der eine Diskettenstation von Vortex hat, aber auch viele Besitzer des Schneider-Laufwerks kennen inzwischen »FILECOPY.COM«, das Dateikopierprogramm von Vortex. Diese komfortable Routine hat dem gleichnamigen Programm auf der Schneider-Systemdiskette klar den Rang abgelaufen.

Während man in der Version von Schneider nur einzelne Dateien oder aber alle gemeinsam kopieren kann, überträgt das deutsche Pendant genau soviele Daten, wie Sie wünschen. Und noch ein Vorteil macht die Arbeit angenehmer. Während bei Schneider die Diskette nach jeder Datei gewechselt werden muß, packt Vortex zuerst den ganzen Speicher voll und »schaufelt« dann die Daten auf die neue Diskette.

Ein Nachteil verblieb bis heute beiden Programmen. Sie arbeiten nicht mit einer Speicherkarte zusammen. Dabei ist FILECOPY von Vortex ganz einfach anzupassen.

Aber nun zur Praxis: Als erstes bespielen Sie eine Diskette mit den CP/M-Systemspuren (Version mit 62 KByte TPA) und den beiden Programmen »DDT.COM« und »FILECOPY.COM«. Arbeiten Sie niemals mit den Originalen, denn Fehler treten schnell auf und CP/M arbeitet so nahe an der Maschinenebene, daß »Abstürze« an der Tagesordnung sind.

Alle Änderungen an der zu patchenden Routine nehmen wir mit dem DDT (Dynamic Debugging Tool) vor. FILECOPY liegt in der Version 2.6 vor. Dieses Programm hat fast jeder zur Verfügung – sofern er nicht zu den Besitzern der allerersten Diskettenstationen gehört. DDT ist ein Debugger mit dem Speicherinhalte untersucht und geändert werden.

Von den insgesamt 13 Anweisungen brauchen wir den S-Befehl. Er wird mit

```
-SAnfangsadresse
```

aufgerufen. Soll ein Wert unverändert bleiben, so drückt man nur die ENTER-Taste. Andernfalls wird der hexadezimale Code des neuen Befehls eingegeben und mit ENTER quittiert. Ein Punkt beendet den Eingabemodus.

Nach dem Start erhalten Sie die Meldung.

```
A>DDT FILECOPY.COM
DDT VERS 2.2
NEXT  PC
0C00 0100
-
```

Mit

```
-S022A
022A FE FE
022B 41 41
022C 28 38
022D 04 F9
022E FE FE
022F 42 44
0230 20 30
0231 F5 F5
0232 C6 .
```

ändern wir die Abfrage der Quell-Diskette. Ab 0242 hex liegt der Block für die Ziel-Diskette.

```
-S0242
0242 FE FE
0243 41 41
0244 28 38
0245 04 F9
0246 FE FE
0247 42 44
0248 20 30
0249 F5 F5
0250 C6 .
```

Als letztes korrigieren wir noch die Texte.

```
-S0905
0905 2F 2D
0906 42 43
0907 29 .
-S092A
092A 2F 2D
092B 42 43
092C 29 .
```

DDT verlassen Sie mit CTRL-C und die neue Datei wird mit

```
A>SAVE 12 COPYFILE.COM
```

gespeichert. Anstatt COPYFILE ist natürlich auch jeder andere Name erlaubt. Nur die Endung ».COM« muß, da es eine Systemdatei ist, vorhanden bleiben.

Wenn wir FILECOPY schon verbessern, dann bauen wir auch gleich noch eine andere Feinheit mit ein. Normalerweise kann man Schneider-Disketten im DATA-Format mit dem Programm von Vortex nicht lesen und damit auch nicht kopieren. Auch das kann mit DDT geändert werden. Dazu brauchen wir den Befehl »A«.

```
-AAnfangsadresse
```

Diese Anweisung ruft den 8080-Assembler auf und die mnemonischen 8080-Befehlswörter können eingegeben werden. Den Eingabemodus verläßt man durch Drücken der ENTER-Taste.

```
A>DDT COPYFILE.COM
```

starten wir und geben als erstes den Befehl

```
-R5
```

ein, um Platz für die neuen Anweisungen zu schaffen. Dann kopieren wir mit

```
-M0105,02A1,0100
```

das Programm und schieben mit

```
-A029D
029D MVI C,0D
029F CALL 0005
02A2
```

die Kommandos zum Zurücksetzen der Diskette ein.

Nun müssen Sie noch den gesamten Bereich zwischen 0203 und 0845 hex auf Sprungbefehle und Startadressen für Textstrings durchforsten. Sofern die Ziele über 029D hex liegen, erhöhen Sie sie jeweils um 5. Die Textstrings werden alle nach folgendem Schema verändert:

```
-l0217
0217 MVI C,09
0219 LXI D,0884
021C CALL 0005
...  ...
-A0219
0219 LXI D,0889
021C
```

Als Ergebnis erscheint

```
-l0217
0217 MVI C,09
0219 LXI D,0889
021C CALL 0005
```

Mit CTRL-C verlassen Sie DDT wie gehabt. Aber vergessen Sie nicht das neue Programm zu speichern.

```
A>SAVE 12 SUPERCOP.COM
```

(Andreas Maurer/hg)



Uhr zurückgedreht

Der Timer der Schneider-Computer eignet sich für vielerlei Anwendungen. Leider ist im Basic aber keine Funktion zum Stellen der Uhr enthalten.

Über »PRINT TIME« kann der Benutzer beim CPC stets die interne Systemzeit abfragen. Leider läßt sich der Timer nicht stellen. Besonders bei der Berechnung von Programmlaufzeiten oder der Auswertung von Reaktionszeiten des Benutzers müßte man den Timer aber eigentlich löschen. Sonst gestaltet sich die Berechnung der vergangenen Zeit schwieriger. Mit einigen POKE-Befehlen funktioniert das auch:

CPC 464:

```
POKE &B187,0:POKE &B188,0:POKE &B189,0:
POKE &B18A,0:PRINT TIME
```

CPC 664 und CPC 6128:

```
POKE &B8B4,0:POKE &B8B5,0:POKE &B8B6,0:
POKE &B8B7,0:PRINT TIME
```

(Martin Kotulla/ja)

POP AF — mal ganz anders

Flags dienen in Maschinencode-Programmen zur Registrierung von Ereignissen. Durch Unterprogramme ändert sich jedoch ihr Inhalt. So stellen Sie den alten Zustand wieder her.

Oft möchte man als Z80-Maschinenprogrammierer auf dem Schneider-CPC den Akku vor dem Aufruf eines Unterprogramms mit PUSH AF sichern und nach dessen Abarbeitung wiederherstellen. Sollen allerdings die Flags, die vom Unterprogramm eventuell verändert wurden, nicht vom folgenden POP AF wieder überschrieben werden, müßte man dazu normalerweise folgende Z80-Befehle benutzen:

```
100 DATA &3E.&C8,&F5,&CD,&2E,&BD,&E3          [66CE]
110 DATA &7C,&E1,&F5,&E1,&22,&60,&01,&C9
                                              [91B0]
120 FOR i=&A000 TO &A00E:READ a               [163A]
130 POKE i,a:NEXT i                           [22F6]
140 MODE 1:PRINT "Der Akku hat den Wert"
;PEEK(&A001)                                  [3278]
150 CALL &A000                                [A0D2]
160 PRINT:PRINT "Aufruf der Maschinenrou
tine":PRINT                                   [096C]
170 PRINT "Der Akku ist immer noch";PEEK
(&161)                                        [2AA2]
180 PRINT "Das Flagregister hat den Wert
";PEEK(&160):PRINT                            [1FD8]
190 IF PEEK(&160) AND 1 THEN PRINT "Druc
ker Busy" ELSE PRINT "Drucker Ready"
                                              [5E24]
200 PRINT                                     [EB7E]
```

Listing. Basic-Demonstration

```
LD   (Speicher),A
CALL Unterprogramm
LD   A,(Speicher)
```

Diese Kommandos retten den Inhalt des Akkumulators in eine Speicherstelle, sichern aber das Flag-Register nicht. Ganz abgesehen davon, daß alleine die Ladebefehle sechs Byte kosten, muß man zudem noch eine zusätzliche absolute Speicheradresse verwalten.

Eine bessere Lösung ist:

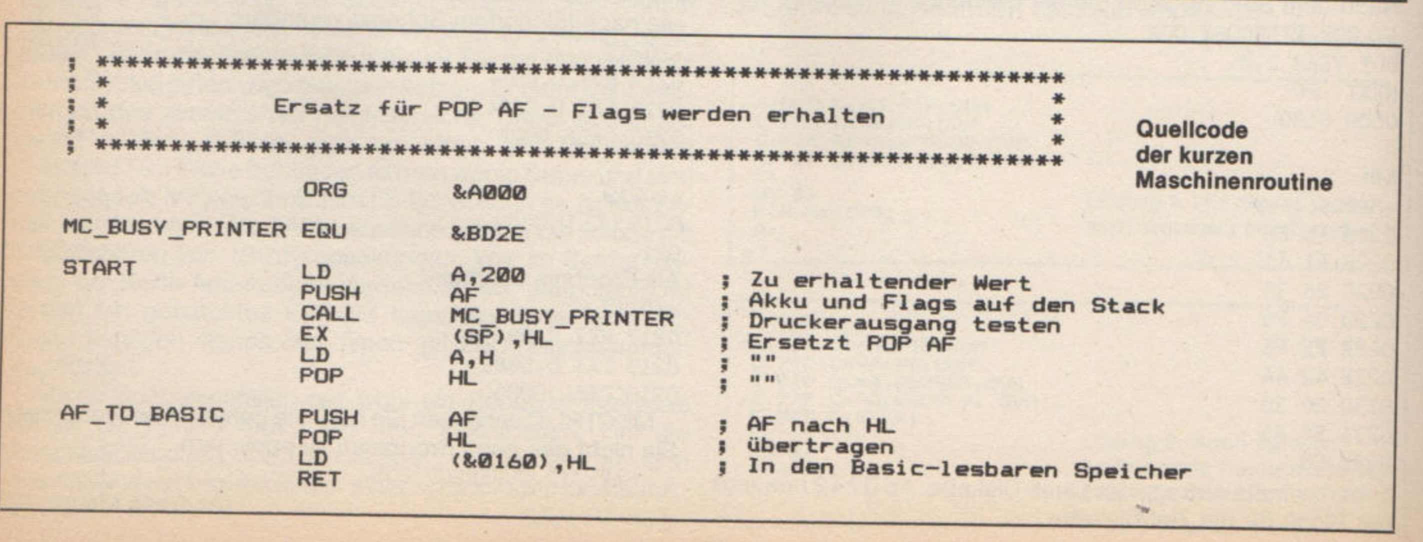
```
PUSH AF
CALL Unterprogramm
EX   (SP),HL
LD   A,H
POP  HL
```

Diese Lösung benötigt nur vier Byte und ist völlig relokatibel. Eine Demonstration zeigt das Listing. Es lädt den Akku mit dem (ansonsten bedeutungslosen) Wert 200 und ruft die CPC-Systemroutine MC BUSY PRINTER auf. Diese setzt das Carry-Flag im F-Register des Prozessors, wenn der Drucker nicht empfangsbereit oder nicht angeschlossen ist. Der Basic-Lader ruft das Maschinenprogramm auf und gibt erläuternde Kommentare aus. Das zeigt, daß die Befehlsfolge den Inhalt des Akkus erhält, aber das Flag-Register sich vom Unterprogramm verändern läßt.

Dieser Trick funktioniert übrigens auf allen Z80-Computern, auch auf dem Joyce. Das Beispielprogramm ist aber auf die CPC-Reihe abgestimmt.

(Martin Kotulla/ja)

Programm-Steckbrief	
Name:	POP AF
Computer:	CPC 464/664/6128
Checksummer:	Explora (für Basic-Lader)
Datenträger:	Kassette/Diskette



```
; ***************************************************************
; *                                                             *
; *     Ersatz für POP AF - Flags werden erhalten               *
; *                                                             *
; ***************************************************************

                  ORG      &A000

MC_BUSY_PRINTER   EQU      &BD2E

START             LD       A,200             ; Zu erhaltender Wert
                  PUSH     AF                ; Akku und Flags auf den Stack
                  CALL     MC_BUSY_PRINTER   ; Druckerausgang testen
                  EX       (SP),HL           ; Ersetzt POP AF
                  LD       A,H               ; ""
                  POP      HL                ; ""

AF_TO_BASIC       PUSH     AF                ; AF nach HL
                  POP      HL                ; übertragen
                  LD       (&0160),HL        ; In den Basic-lesbaren Speicher
                  RET
```

Quellcode der kurzen Maschinenroutine

Vier KByte gratis

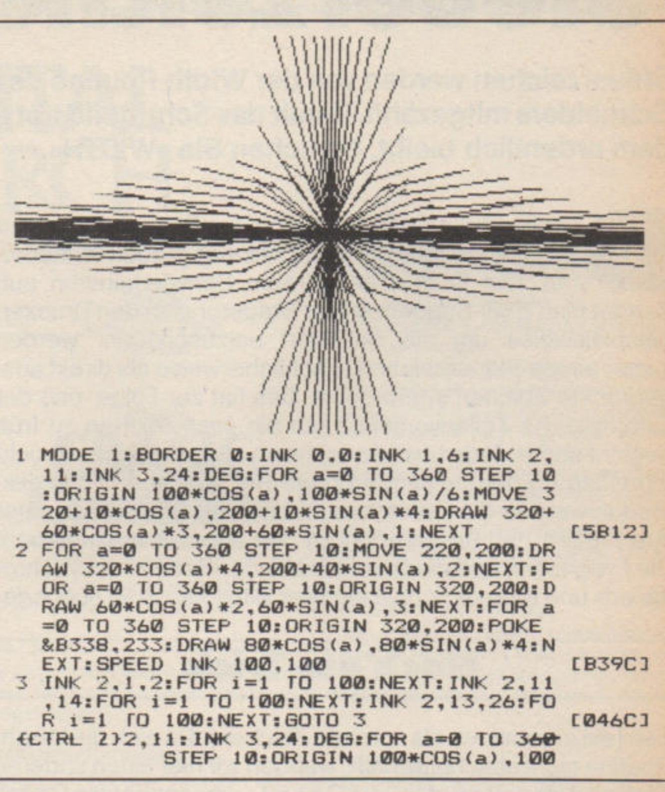
Das Arbeiten mit sequentiellen Dateien beim CPC 464 ist zeitraubend. Eine kurze Maschinenroutine schafft Abhilfe und zusätzlichen Speicherplatz auf unkonventionelle Weise.

Bei jedem sequentiellen Datenzugriff mit »OPENIN« und »OPENOUT« legt der CPC einen Puffer an. Dieser ist vier KByte lang und liegt direkt unter HIMEM. Die hier ansässigen Stringvariablen müssen vorher unter viel Zeitaufwand verschoben werden. Das Zeitproblem kann man zwar mit der Formel »OPENIN "dummy":MEMORY HIMEM-1 :CLOSEIN« abfangen, wenn Sie jedoch Wert auf einen eigenen Zeichensatz und andere Erweiterungen legen, können Speicherplatzprobleme auftreten.

Das Programm »Sequbild« verlegt den Puffer in den Bildschirmspeicher und vermeidet so elegant jedes Debakel. Das erfordert zwei Maschinencode-Routinen, die den Routinen »OPENIN« und »OPENOUT« entsprechen. Nach dem Einlesen ersetzen Sie in Ihrem Programm alle »OPENIN« durch »CALL HIMEM+1« und alle »OPENOUT« durch »CALL HIMEM+13«. Eine Eingabedatei wird mit »CALL &BC7A« geschlossen, eine Ausgabedatei mit »CALL &BC8F«. Das eigentliche Programm finden Sie in den Zeilen 10090 bis 10180. Den anschließenden Demonstrationsteil benötigen Sie bei der Verwendung in eigenen Programmen natürlich nicht.

Beachten Sie, daß der Zugriff auf sequentielle Dateien den Bildschirminhalt zerstört. Während des Ladens von Diskette (oder Kassette) darf der Bildschirminhalt nicht geändert werden.

(Dieter Braun/Matthias Rosin)



```
1 MODE 1:BORDER 0:INK 0.0:INK 1.6:INK 2,
  11:INK 3,24:DEG:FOR a=0 TO 360 STEP 10
  :ORIGIN 100*COS(a),100*SIN(a)/6:MOVE 3
  20+10*COS(a),200+10*SIN(a)*4:DRAW 320+
  60*COS(a)*3,200+60*SIN(a),1:NEXT          [5B12]
2 FOR a=0 TO 360 STEP 10:MOVE 220,200:DR
  AW 320*COS(a)*4,200+40*SIN(a),2:NEXT:F
  OR a=0 TO 360 STEP 10:ORIGIN 320,200:D
  RAW 60*COS(a)*2,60*SIN(a),3:NEXT:FOR a
  =0 TO 360 STEP 10:ORIGIN 320,200:POKE
  &B338,233:DRAW 80*COS(a),80*SIN(a)*4:N
  EXT:SPEED INK 100,100                     [B39C]
3 INK 2,1,2:FOR i=1 TO 100:NEXT:INK 2,11
  ,14:FOR i=1 TO 100:NEXT:INK 2,13,26:FO
  R i=1 TO 100:NEXT:GOTO 3                  [046C]
{CTRL Z}2,11:INK 3,24:DEG:FOR a=0 TO 360
          STEP 10:ORIGIN 100*COS(a),100
```

Programm-Steckbrief	
Name:	Sequbild
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
10000 '********************************
                                        [4E44]
10010 '*  sequenzieller Datenzugriff   *
                                        [917C]
10020 '* ueber den Bildschirmspeicher  *
                                        [2C98]
10030 '*                               *
                                        [1FDE]
10040 '*      (c) by Dieter Braun      *
                                        [F0A8]
10050 '*                               *
                                        [4FE2]
10060 '*        November 1985          *
                                        [AE0E]
10070 '********************************
                                        [3652]
10080 '                                 [3C80]
10090 '  Daten des MC-Programmes einlese
      n                                 [346C]
10100 '                                 [1672]
10110 MODE 2                            [F914]
10120 h=HIMEM:MEMORY HIMEM-24:s=0       [08F2]
10130 DATA &21,&0,&C0,&46,&23,&11,&0,&C1
      ,&CD,&77                          [924E]
10140 DATA &BC,&C9,&21,&0,&C0,&46,&23,&1
      1,&0,&C1,&CD,&8C,&BC,&C9          [B5FE]
10150 FOR n=HIMEM+1 TO HIMEM+24         [15F2]
10160 READ a:POKE n,a:s=s+a             [3744]
10170 NEXT n                            [68CC]
10180 IF s<>2527 THEN PRINT"Pruefsummenf
      ehler in den Datas{2 SPACE}!!":END
                                        [E282]
10190 '                                 [DF84]
10200 '  Dateiname eingeben + auf Laenge
       pruefen                          [650E]
10210 '                                 [0F76]
10220 INPUT "Dateiname";a$              [FEA2]
10230 IF LEN(a$)>12 THEN 10220          [34FE]
10240 CLS                               [A6F2]
10250 GOSUB 10510:' Dateiname ablegen   [F14C]
10260 '                                 [3680]
10270 '  Datei anlegen                  [76C4]
10280 '                                 [3884]
10290 CALL HIMEM+13:'Datei eroeffnen,
            entspricht 'openout a$'     [D788]
10300 PRINT#9,"Datenzugriff"            [1BAC]
10310 PRINT#9,"ueber den Bildschirmspeic
      her"                              [5198]
10320 CALL &BC8F:' Datei wieder schliess
      en,   entspricht 'closeout'       [4D40]
10330 '                                 [117C]
10340 ' Dateiname nochmals im Bildschirm
      speicher ablegen                  [D068]
10350 '                                 [3780]
10360 CLS                               [B4F8]
10370 GOSUB 10510:' Dateiname nochmales
      ablegen                           [D506]
10380 '                                 [3A86]
10390 '  Datei wieder einlesen          [19FC]
10400 '                                 [3C78]
10410 CALL HIMEM+1:' Datei eroeffnen,
            entspricht 'openin a$'      [B654]
10420 LINE INPUT #9,a$                  [6838]
10430 LINE INPUT #9,b$                  [4F3C]
10440 CALL &BC7A:' Datei wieder schliess
      en,   entspricht 'closein'        [E238]
10450 '                                 [1782]
10460 MODE 2:PRINT:PRINT:PRINT:PRINT "In
      halt der Datei :":PRINT:PRINT a$  [9EC2]
10470 PRINT:PRINT b$:PRINT:PRINT:END    [156A]
10480 '                                 [E488]
10490 ' Unterprogramm  Dateiname an Anfa
      ng des Bildschirmspeichers poken  [F5F4]
10500 '                                 [3E7A]
10510 POKE 49152,LEN(a$):REM Laenge able
      gen                               [3688]
10520 FOR n=1 TO LEN(a$)                [B826]
10530  POKE 49152+n,ASC(MID$(a$,n,1))   [168A]
10540 NEXT                              [A6B2]
10550 RETURN                            [D2F6]
```

Listing. »Sequbild erschließt viel Speicherplatz mit wenig Programm«

Steuerzeichen austricksen

Steuerzeichen werden von der Width-Routine des Schneiders mitgezählt. Damit das Schriftbild trotzdem ordentlich bleibt, brauchen Sie »WIDTH«.

Die eingebaute Width-Routine des Schneiders ist nur für Listings geeignet. Möchte man jedoch formatierten Text ausdrucken, treten Schwierigkeiten auf: Sendet man ESC-Sequenzen zur Steuerung an den Drucker, beispielsweise um die Schriftart umzuschalten, werden immer einige Steuerzeichen fälschlicherweise als direkt ausgedruckte Zeichen interpretiert. Das hat zur Folge, daß der automatische Zeilenvorschub um ein paar Zeichen zu früh beginnt und so jede noch so mühsam ausgeknobelte Druckformatierung durcheinanderbringt. Bei Ausgaben im Grafik-Modus werden die Zeilenvorschubbefehle als Bitbildmuster interpretiert und unverändert an den Drucker ausgegeben. Die Programm- und die Druckerzählung werden »desynchronisiert« und so zum Schluß ein paar Bitbilder zuviel ausgegeben (sie erscheinen als normale Zeichen). Da sich die Width-Routine nicht abschalten läßt, wenden wir hier einen anderen Trick an: Der neue Befehl »ESC,x1,x2,...xn« sendet die Codes x1 bis xn ohne Zählung durch die Width-Routine an den Drucker. Dabei ist die Parameterzahl beliebig (zwischen 0 und 32). Außerdem erspart man sich gegenüber »PRINT #8,...« das lästige »CHR$(x)«. Es genügt, x allein anzugeben.

Noch ein Trick

Und so setzen Sie die Befehlserweiterung ein:

Geben Sie einfach das angegebene Listing ein und starten Sie es durch »RUN«. Der im Programm vorhandene Call-Befehl bindet die Befehlserweiterung in das Betriebssystem ein, so daß der Aufruf durch »ESC,x1,x2,...xn« erfolgen kann. Das Ladeprogramm dürfen Sie anschließend mit »NEW« löschen, ohne daß die Befehlserweiterung verlorengeht.

(Helmut Tischer/hg)

Programm-Steckbrief	
Name:	Width
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
100 PRINT"Wieviel Zeichen sollen frei de
    finierbar sein?"                        [42B2]
110 INPUT "SYMBOL AFTER ",sym%              [B9B0]
120 SYMBOL AFTER sym%                       [D272]
130 OPENOUT"dummy" 'Kassettenpuffer auf
    den Bereich oberhalb RAMTOP fixieren
                                            [C240]
140 MEMORY HIMEM-1:CLOSEOUT                 [B448]
150 SPEED WRITE 1                           [A346]
180 READ n,m:st=HIMEM-n+1:MEMORY st-1       [3272]
190 FOR i=0 TO n-1                          [258C]
200 READ d$:d=VAL("&"+d$):POKE st+i,d       [F36C]
210 NEXT                                    [6CE4]
220 FOR i=1 TO m                            [78C4]
230 READ d$                                 [9AF2]
240 d=VAL("&"+d$)                           [FF50]
250 z=st+PEEK(st+d)+256*PEEK(st+d+1)        [B54C]
260 POKE st+d+1,INT(z/256)                  [D20E]
270 POKE st+d,z-256*INT(z/256)              [DF34]
280 NEXT                                    [70F2]
310 CALL HIMEM+1                            [9978]
320 END                                     [9B18]
360 DATA 51,6                               [2776]
380 DATA 01,09,00,21,12,00,c3,d1            [3748]
390 DATA bc,0e,00,c3,16,00,45,53            [2024]
400 DATA c3,00,00,00,00,00,a7,c8            [BD96]
410 DATA 47,dd,e5,e1,87,5f,16,00            [40AE]
420 DATA 19,2b,2b,7e,cd,2a,00,10            [2CDA]
430 DATA f8,c9,cd,30,00,30,fb,c9            [6460]
440 DATA cf,f2,87                           [C834]
450 'Ziffern der zu aendernden Adressen     [DB4E]
460 DATA 0001,0004,0009,000c                [E472]
470 DATA 0025,002b                          [0EB8]
```

Listing. Überlisten Sie den Zeilenvorschub

Modi-Check

Wenn Sie bei einem Programm feststellen wollen, in welchem Modus sich Ihr Bildschirm befindet, läßt Ihr Basic Sie im Stich.

Oft ist es ganz nützlich, zu erfahren, welcher Bildschirm-Modus im Schneider CPC gerade aktiv ist. Mit PEEK ist das an sich kein Problem. Doch wenn ein Programm auf allen drei Schneider CPCs laufen soll, sind verschiedene Adressen zuständig: Beim CPC 464 heißt es »PRINT PEEK(&B1C8)«, bei den Modellen CPC 664 und CPC 6128 hingegen »PRINT PEEK(&B7C3)«.

Eine bessere, da versionsunabhängige, Lösung finden Sie im Listing. Der Computer gibt einen Wagenrücklauf an den Bildschirm, der den Cursor an die erste Position der Zeile setzt. Ein Backspace (ASCII 8) weist das Betriebssystem an, den Cursor auf die letzte Stelle der vorhergehenden Zeile zu positionieren. Über die Basic-Funktion POS wird die Cursorspalte abgefragt. Es kann hier (wenn Sie nicht gerade Windows verwenden) nur die Werte 20, 40 und 80 geben – entsprechend dem gewählten Modus.

(Martin Kotulla/ja)

Programm-Steckbrief	
Name:	Modi-Tester
Computer:	CPC 464/664/6128
Checksummer:	Explora
Datenträger:	Kassette/Diskette

```
100   MODE bestimmen ************           [9232]
110 a=POS(#0):b=VPOS(#0)                    [5A56]
120 PRINT CHR$(13);CHR$(8);                 [862C]
130 x=POS(#0)                               [1DFE]
140 LOCATE a,b                              [98F8]
150 IF x=20 THEN PRINT "MODE 0"             [1CA2]
160 IF x=40 THEN PRINT "MODE 1"             [34AA]
170 IF x=80 THEN PRINT "MODE 2"             [36B6]
180 END                                     [A120]
```

Immer im Bilde über den Modus

Leistungsfähige Programmiersprachen für Schneider CPC 464/664/6128 + Joyce



Pascal/MT+

Eine der umfangreichsten Pascal-Implementationen für 8-Bit-Mikrocomputer. Pascal/MT+ ist ein volles ISO-Standard-Pascal, das um eine leistungsfähige Programmierumgebung für Industrie-, Geschäfts- und Ausbildungs-Einsatz sowie Möglichkeiten zur Systemprogrammierung erweitert wurde.

Pascal/MT+ erweitert die bekannten Vorteile der strukturierten Sprache Pascal. Es ist schneller, vielseitiger, portabler und in anspruchsvollen Anwendungen, die die Entwicklung separater Programm-Module erfordern, einfacher zu verwenden.

Markt&Technik
Schneider CPC-Software
DIGITAL RESEARCH
Pascal MT+™
FROM DIGITAL RESEARCH THE CREATORS OF CP/M
3"-Diskette für CPC 464*, 664*, 6128, Joyce PCW 8256
Handbuch in englischer Sprache

Direkte Umsetzung in schnellen Objekt-Code

Im Unterschied zu Compilern, die in einen Zwischencode übersetzen, wandelt Pascal/MT+ direkt in schnellen Objekt-Code um. Die Ausführungszeiten sind deshalb wesentlich besser als bei traditionellen Pseudo-Code-Compilern.

Das Pascal/MT+-Paket beinhaltet:

- einen Compiler, der relokatierbare Objekt-Dateien erzeugt,
- einen Linker, der lauffähige Programme erzeugt,
- eine Laufzeitbibliothek
- einen Disassembler, der die Untersuchung des erzeugten Codes ermöglicht, und
- einen Debugger, der einen symbolischen Test eines Programms erlaubt

Die Bibliothek enthält Routinen von der Berechnung transzendenter Funktionen bis zur Verwendung von Maschinen-Interrupten.

Ideal für Geschäfts-, Industrie- und Ausbildungs-Einsatz

Zusätzlich zu den numerischen Standard-Datentypen unterstützt Pascal/MT+ entweder Fließkommazahlen oder binär-codierte Dezimalzahlen (BCD) und erzielt damit die in kommerziellen Anwendungen so wichtige Genauigkeit von Ergebnissen ohne Rundungsfehler.

Für industrielle Anwendungen bietet Pascal/MT+ den Vorteil von ROM-fähigem Maschinencode, Möglichkeiten zur Reduzierung der Programmgröße und erweiterte Ein-/Ausgabefähigkeiten.

Für den Einsatz im Ausbildungs- und Lehrbereich empfiehlt sich Pascal/MT+, weil es eine volle Implementation von Pascal ist, die man leicht erlernen kann, aber später auch bei gestiegenen Ansprüchen noch leistungsfähig ist.

Genügt professionellen Ansprüchen

Pascal/MT+ wurde für die hohen Ansprüche professioneller Softwareentwickler und erfahrener Anwender entwickelt. Pascal/MT+ wird mit ausführlicher Dokumentation in englischer Sprache geliefert.

Hardwarevoraussetzungen

Pascal/MT+ läuft auf den Schneider-Computern CPC 464 und CPC 664 (mit Speichererweiterung), dem CPC 6128 und dem PCW 8256 (Joyce) unter CP/M und CP/M-Plus. Kompilierte Programme sind, bei entsprechender Größe, auch auf dem CPC 464 und CPC 664 ohne Speichererweiterung lauffähig.

Die Vorteile von Pascal/MT+ auf einen Blick:

- Superset des ISO-Standard-Pascal
- Kompilierung separater Module
- erzeugt effektiven Maschinencode
- komplette Entwicklungstools
- erweiterte Datentypen (BYTE, WORD, LONGINT, STRING)
- Bit- und Byte-Manipulationen
- schneller Dateizugriff
- Direktzugriffsdateien
- CHAINing mit Übergabe von Variablen zwischen Overlays
- umfangreiche Dienstprogramme

Best.-Nr. MS 611 **DM 174,–*** (sFr. 158,–/öS 1680,–*)



CBASIC-Compiler

Der Hochleistungs-BASIC-Compiler für Softwareprofis zur Erstellung kommerzieller Anwendungen.

Der CBASIC-Compiler ist ein erweitertes BASIC mit wichtigen Vorteilen für Softwareprofis. Er ist ein Compiler, der Maschinencode erzeugt und die Programmierung und den Test separater Module erlaubt, die später ein komplettes Programm ergeben sollen. Die integrierten Grafikmöglichkeiten des CBASIC-Compilers erlauben die Programmierung vielseitiger Grafikprogramme für eine Vielzahl von Anwendungen (nur auf Computern mit GSX-Software).

Schnelle Ausführung

Der CBASIC-Compiler kombiniert die Geschwindigkeit von Maschinencode mit der leichten Verständlichkeit der Sprache BASIC. Ein mit dem CBASIC-Compiler kompiliertes Programm wird acht- bis zehnmal schneller ausgeführt als das gleiche interpretierte Programm.

Grafikerweiterungen

Der CBASIC-Compiler beinhaltet einen voll integrierten Satz von Grafikbefehlen und -funktionen. Geräteunabhängige Grafikfähigkeiten ermöglichen die Ausgabe von Grafiken auf jedem unterstützten Grafikausgabegerät (Bildschirm, Drucker, Plotter) ohne Neukompilierung eines Programms.

Dezimal-Arithmetik

Die 14stellige Dezimal-Arithmetik gewährleistet höchste Genauigkeit bei Berechnungen und stellt sicher, daß alle Geldbeträge auf den Pfennig genau stimmen. Rundungsfehler, wie sie bei binärer Arithmetik möglich sind, können nicht auftreten.

Der CBASIC-Compiler unterstützt auch echte Integer-Arithmetik, so daß zur Erhöhung der Geschwindigkeit auch Integer-Variablen verwendet werden können.

Mehrzeilige Funktionen

Durch die Möglichkeit, mehrzeilige Funktionen zu erstellen, verfügt der CBASIC-Compiler über Fähigkeiten, die sich sonst nur in strukturierten Programmiersprachen wie PL/I oder Pascal finden. Innerhalb einer mehrzeiligen Funktion können lokale Variablen verwendet werden.

Für professionellen Einsatz

Der CBASIC-Compiler wurde für die hohen Ansprüche professioneller Softwareentwickler und erfahrener Anwender entwickelt. Der CBASIC-Compiler wird mit ausführlicher Dokumentation in englischer Sprache geliefert.

Hardwarevoraussetzungen

Der CBASIC-Compiler läuft auf Schneider CPC 464 mit Diskettenlaufwerk DDI-1, dem CPC 664, dem CPC 6128 und dem 8256 (Joyce). Für Grafikprogramme wird die GSX-Software benötigt, die nur mit dem CPC 6128 und PCW 8256 (Joyce) ausgeliefert wird. Die Grafiken können dann auf dem Bildschirm oder einem von GSX unterstützten Drucker oder Plotter ausgegeben werden. Es können zum Beispiel der NLQ 401-Matrixdrucker, ein Epson- oder kompatibler Drucker und HP- und HP-kompatible Plotter zur Ausgabe verwendet werden.

Die Vorteile des CBASIC-Compilers auf einen Blick:

- hohe Geschwindigkeit der erzeugten Programme
- Grafikerweiterungen
- Dezimal-Arithmetik mit hoher Genauigkeit
- umfangreiche Stringverarbeitung
- Stringlänge bis 32 KByte
- mehrzeilige Funktionen
- keine Zeilennummern erforderlich
- Overlays durch CHAIN-Befehl

Best.-Nr. MS 612 **DM 174,–*** (sFr. 158,–/öS 1680,–*)

* inkl. MwSt. Unverbindliche Preisempfehlung



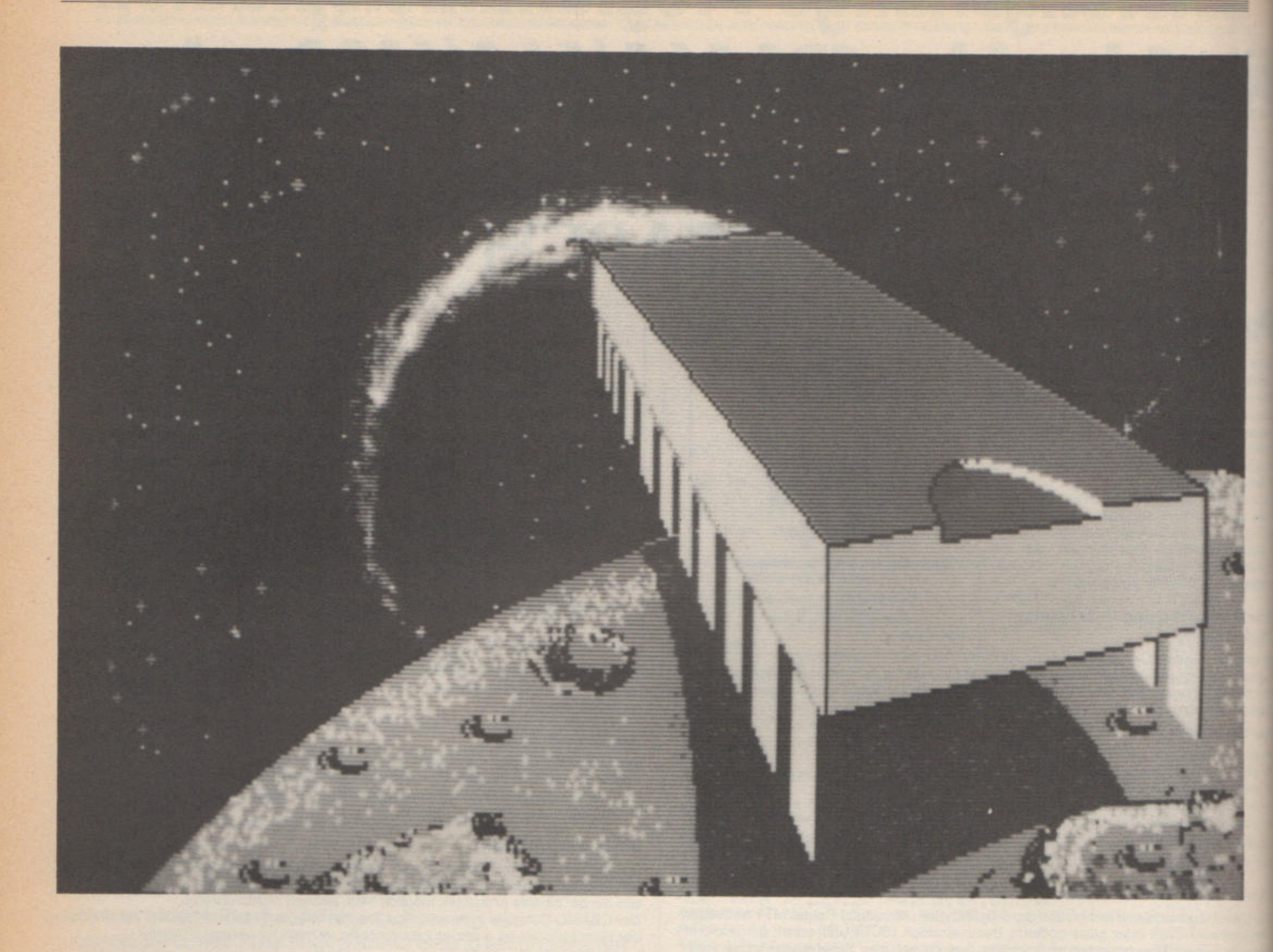
Unternehmensbereich Buchverlag
Hans-Pinsel-Straße 2, 8013 Haar bei München

Diese Markt&Technik-Softwareprodukte erhalten Sie in den Fachabteilungen der Kaufhäuser und in Computershops.

Bestellungen im Ausland bitte an untenstehende Adressen.

Schweiz: Markt&Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel. 042/41 56 56

Österreich: Ueberreuter Media Handels- und Verlagsges. mbH, Alser Straße 24, A-1091 Wien, 0222/48 15 38-0



CP/M mit Nachbrenner

CP/M Plus ist der Nachfolger von CP/M 2.2. Komfort wird bei diesem – auch CP/M 3.0 genannten – Betriebssystem groß geschrieben. Vorausgesetzt, man weiß mehr, als im Handbuch steht.

CP/M Plus hat erst seit dem Erscheinen des Schneider CPC 6128 und des Commodore 128 breite Verwendung gefunden. Es ist ebenso wie der Vorgänger CP/M 2.2 ein Standard-Betriebssystem. Die Bedienung wurde aber um ein Vielfaches komfortabler. Und auch die Programmierhilfen erfuhren eine wesentliche Verbesserung. Gleiches gilt für die inzwischen bedeutend aufschlußreicheren Fehlermeldungen. Das ganze System nimmt dem Benutzer Fehler bei weitem nicht mehr so übel wie CP/M 2.2.

CP/M Plus stammt aus dem Jahr 1982 und war der letzte Versuch von Digital Research, den vordringenden 16-Bit-Geräten der IBM-PC-Klasse mit der bewährten 8-Bit-Technologie Paroli zu bieten. Die bedeutendste Eigenschaft dieses hochentwickelten Betriebssystems ist die Fähigkeit, mehr als 64 KByte RAM-Speicher ansprechen zu können. CP/M 2.2 hingegen ist auf ein Maximum von 64 KByte hin konzipiert. Doch die Verwaltung des großen Speichers ist nur beschränkt möglich. Den Ausbau – beispielsweise auf 512 KByte – unterstützt CP/M Plus zwar, aber aufgrund der Adressierungseigenschaften des Z80-Prozessors dürfen die Programme trotzdem nur maximal 64 KByte lang sein, wenn sie in einem Stück in den Speicher passen sollen. Der restliche Speicher kann nur als RAM-Disk, Druckerspooler oder Diskettenpuffer benutzt werden.

Trotzdem holt CP/M Plus alles Machbare aus dem 8-Bit-Computer heraus. Dennoch konnte sich bisher die Version 3.0 nicht durchsetzen. Aber mit dem CPC 6128 von Schneider und dem Commodore 128 hat sich die Situation geändert. Diese beiden Computer sind Massenprodukte für Privatleute sowie das »Small-Business« und arbeiten nur »nebenbei« auch unter CP/M 3.0.

Je nach Computertyp stehen unter CP/M 2.2 bis zu 60 KByte freier Arbeitsspeicher zur Verfügung. Beim Schneider sind es mit diesem Betriebssystem nur rund 39 KByte RAM für die TPA (Transient Program Area). CP/M 3.0 bietet immerhin 61 KByte. Damit sind viele Standardprogramme erst voll funktionsfähig, die bei einer kleinen TPA nur mit Einschränkungen oder nach Einbau einer Speichererweiterung zum Laufen gebracht werden konnten.

Auf Grund der Verbesserungen erhöhte sich natürlich auch der Umfang des Programmcodes von CP/M Plus wesentlich. Kam CP/M 2.2 noch mit einem Minimal-RAM von 20 KByte aus, so verlangt 3.0 mindestens 96 KByte Speicherplatz. Einer speziellen Version von Digital Research reicht allerdings eine einzige 64-KByte-Bank. Sie braucht dennoch erheblich mehr Speicherplatz als CP/M 2.2. Dieser muß nun von der TPA »abgezwackt« werden. Fer-

ner bietet diese Sparversion des CP/M Plus nicht alle Leistungsmerkmale des kompletten CP/M Plus-Betriebssystems, so daß sie auf fast keinem Computer (Selbstbau ausgenommen) installiert wurde.

Schneider benutzt die vollständige Version von CP/M 3.0 mit mehreren Speicherbänken. So findet nicht mehr das gesamte Betriebssystem auf den Systemspuren der Diskette Platz. Denn auf zwei Spuren mit neun Sektoren passen 9216 (=2*9*512) Byte. Zu

Besser und länger

wenig für CP/M Plus, deren Datei C10CPM3.EMS, in der das Betriebssystem jetzt steht, schon 25 KByte umfaßt.

In den Systemspuren befindet sich deshalb lediglich ein Ladeprogramm, das die Datei C10CPM3.EMS in den Speicher verfrachtet. Das »EMS« der Extension des Dateinamens steht übrigens für »Early Morning-Startup« – ein typisches Beispiel für britischen Humor. Ohne diese EMS-Datei kann CP/M Plus nicht gestartet werden. Deshalb fallen von den auf einer Diskette zur Verfügung stehenden 169 KByte weitere 25 KByte für diese Datei weg, so daß die 3-Zoll-Disketten, vom Speichervolumen her betrachtet, recht schlecht dastehen.

Ein Trostpflaster ist, daß die EMS-Datei nur zum Systemstart gebraucht wird, also sich keinesfalls auf jeder CP/M Plus-Diskette befinden muß. Zudem hat CP/M Plus auch gegen das Data-Only-Format nichts einzuwenden, das 178 KByte formatierte Speicherkapazität bietet.

CP/M Plus wird – wie auch CP/M 2.2 – durch Eingabe des RSX-Befehls »|CPM« gestartet. Der Computer erkennt selbständig anhand der eingelegten Diskette, ob er CP/M 2.2 oder CP/M Plus laden soll. Das Booten des Betriebssystems dauert bei CP/M 3.0 deutlich länger als bei CP/M 2.2. Verständlich, denn der Programmcode wurde schließlich auch erheblich umfangreicher.

Die CP/M-Startmeldung lautet:

```
CP/M Plus Amstrad Consumer
Electronics plc
v 1.0, 61K TPA, 1 disc drive
```

Die Startmeldung zeigt, wie viele Diskettenlaufwerke an den Computer angeschlossen sind. Falls Sie trotz Zweitstation die Meldung »1 disc drive« erhalten, haben Sie wahrscheinlich vergessen, beim Booten des Betriebssystems eine Diskette in diese Station zu legen und die Laufwerksklappe zu schließen.

Die Feststellung, ob ein oder zwei Diskettenlaufwerke benutzt werden, hat durchaus einen tieferen Sinn. Fehlt die zweite Station, kann das Betriebssystem diese nämlich simulieren. Sobald von einem laufenden Programm oder vom Benutzer ein Zugriff auf das (nicht vorhandene) zweite Laufwerk verlangt wird, erscheint in der untersten Bildschirmzeile folgende Laufschrift:

```
Please put the disc for B:
into the drive then press any key
```

Sie legen dann die B-Diskette ins A-Laufwerk ein und das Betriebssystem »gaukelt« dem Anwenderprogramm vor, daß die Daten von der imaginären Diskettenstation kommen.

Damit Sie aber wissen, welches Laufwerk gerade selektiert ist, steht in der rechten unteren Bildschirmecke die Meldung »Drive is A:« oder »Drive is B:«. So können Sie viele Programme, die eigentlich mit zwei Laufwerken arbeiten, auch mit nur einer Diskettenstation verwenden. Das ständige Wechseln von Disketten ist auf Dauer allerdings etwas mühselig. Und auch die Mechanik des Disketten-Laufwerks nimmt die Doppelbelastung auf längere Sicht eventuell übel.

Neben dieser Hilfsroutine, über deren Sinn man durchaus geteilter Meinung sein kann, existieren noch diverse Mittel, die die Arbeit mit CP/M Plus sehr vereinfachen. Unter CP/M 2.2 ist das Editieren von Befehlszeilen bekanntlich eine Qual. Außer DEL – beziehungsweise CTRL-H – gibt es kaum eine Korrekturmöglichkeit der vorgenommenen Eingaben. CP/M Plus hat da deutlich zugelegt und erreicht beinahe den Komfort des Basic-Editors eines Schneider CPC.

Am besten arbeiten Sie jetzt am Computer mit, um sofort die Auswirkungen der einzelnen Befehle zu sehen. Sie brauchen dazu eine Kopie der Diskette 1 des CP/M-Betriebssystems. Mit dem RSX-Befehl »|CPM« starten Sie CP/M 3.0. Tippen Sie bitte ein:

```
A>POP CON:=KRYS.WP(NZ)
```

Editieren mit Komfort

In diese Zeile haben sich nun eine ganze Reihe von Fehlern eingeschlichen. So muß die Angabe »NZ« in eckigen Klammern stehen – nicht in runden. Um das letzte Zeichen auszubessern, können Sie einfach CTRL-H oder DEL drücken. Dann steht auf dem Bildschirm:

```
A>POP CON:=KRYS.WP(NZ
```

Nun fügen Sie die eckige Klammer an:

```
A>POP CON:=KRYS.WP(NZ]
```

Wenn Sie jetzt viermal CTRL-A drücken, wandert der Cursor nach links über das Zeichen »(«. CTRL-G löscht dieses und zieht den Rest der Zeile von rechts heran:

```
A>POP CON:=KRYS.WPNZ]
```

Sie müssen hier nun auch eine eckige Klammer einsetzen:

```
A>POP CON:=KRYS.WP[NZ]
```

Als nächstes nehmen wir uns »POP« vor, wofür eigentlich »PIP« stehen soll. Tippen Sie CTRL-B, und der Cursor springt an den Anfang der Eingabezeile. Einmal CTRL-F gedrückt, und der Cursor bewegt sich eine Position nach rechts auf das »O«. Wieder löscht CTRL-G das betroffene Zeichen und »I« wird eingesetzt:

```
A>PIP CON:=KRYS.WP[NZ]
```

Jetzt sind Sie an der Reihe: Ändern Sie »KRYS.WP« in »KEYS.WP« ab. Sie wissen doch noch, wie man den Cursor bewegt und Zeichen löscht? Als Resultat muß jedenfalls

```
A>PIP CON:=KEYS.WP[NZ]
```

auf dem Bildschirm stehen.

Wenn Ihnen nach der Ausführung des Befehls die automatische Zeilennumerierung von PIP.COM mit der »N«-Option nicht gefällt, können Sie das ändern, ohne die Zeile noch einmal einzugeben.

Die Geheimnisse der Control-Codes

Drücken Sie einfach CTRL-W, worauf der Computer die Zeile noch einmal so anzeigt, als hätten Sie sie gerade eingegeben. Mit den bekannten Editierkommandos ist es ein leichtes, die Zeile in

```
A>PIP CON:=KEYS.WP[Z]
```

abzuändern.

Zur Übersicht finden Sie im folgenden alle CTRL-Codes, die Ihnen der CCP (Console Command Processor) zum Editieren von Befehlszeilen anbietet:

CTRL-A bewegt den Cursor um eine Bildschirmposition nach links.

CTRL-B stellt den Cursor an den Anfang der Eingabezeile. Befindet er sich bereits dort, springt er an das Ende der Zeile.

CTRL-E fügt einen physikalischen Wagenrücklauf ein, der von CP/M aber nicht beachtet wird. So dürfen Sie statt der obigen Zeile gleichwertig auch schreiben (»^« steht für die CTRL-Taste):

```
A>PIP CON:=KEY^E
S.WP[Z]
```

CTRL-F schiebt den Cursor um ein Zeichen nach rechts.

CTRL-G löscht das unter dem Cursor stehende Zeichen.

CTRL-H und **DEL** entfernt das Zeichen links vom Cursor und bewegt den Cursor um dieses eine Zeichen nach links.

CTRL-I und **TAB** positioniert den Cursor am nächsten Tabulatorstop. Eventuell rechts vom Cursor stehende Zei-

chen werden ebenfalls nach rechts verschoben.

CTRL-J, **CTRL-M**, **ENTER** und **RETURN** schließen eine Eingabezeile ab, die in der Folge vom System interpretiert wird.

CTRL-K entfernt alle Zeichen von der Cursorposition bis zum Ende der Eingabezeile.

CTRL-R schreibt einen eingegebenen Befehl nochmals auf den Bildschirm, wo Sie ihn bequem editieren können. Dieser Steuercode ist nur wirksam, bevor Sie RETURN drücken.

CTRL-S stoppt das Bildschirm-Scrollen.

CTRL-Q nimmt es wieder auf.

CTRL-U behält seine alte Aufgabe aus CP/M 2.2 und ist dazu gedacht, die Bearbeitung einer Eingabezeile abzubrechen. Es erscheint das Doppelkreuz »#«, das Sie auffordert, Ihre Eingabe mit den gewünschten Korrekturen zu wiederholen.

CTRL-W wiederholt eine vorherige Eingabe. Wenn Sie schon Zeichen in der Zeile eingetippt haben, stellt sich der Cursor lediglich an das Ende der Eingabezeile.

CTRL-X löscht alle Zeichen vom Zeilenanfang bis zur Cursorposition und zieht den rechtsstehenden Text nach links.

Die weiteren Control-Funktionen sind den meisten CP/M-Benutzern bekannt. Da gibt es CTRL-C, mit dem man unter der Version 2.2 jede Diskette, die in einem Laufwerk gewechselt wurde, anmelden muß. CP/M Plus ist so anwenderfreundlich, daß es bei jedem Zugriff ein »Log-In« durchführt, wodurch das lästige Anmelden der Disketten vollständig entfällt.

Wenn Sie **CTRL-C** drücken, löst der Computer aber weiterhin einen Warmstart aus. Sie werden dabei bemerken, daß das Diskettenlaufwerk nicht anläuft, um das BDOS und den CCP nachzuladen. Beide Systemteile sind unter 3.0 in der Systembank der Speichererweiterung abgelegt und können bei Bedarf blitzschnell zurückkopiert werden.

Auch **CTRL-P** erfuhr eine Verbesserung. Diese Tastenkombination schaltet das automatische Mitprotokollieren von Bildschirmausgaben auf dem Drucker ein und wieder aus. Neu ist, daß ein nicht angeschlossener (beziehungsweise nicht empfangsbereiter) Drucker nicht mehr zu einem Systemabsturz führt. Nach einigen Sekunden erscheint vielmehr in der untersten Bildschirmzeile die Meldung:

```
LPT not ready - Retry,
Ignore or Cancel?
```

»R« startet einen neuen Versuch zur Druckerausgabe, »I« ignoriert, daß ein Zeichen nicht abgesetzt werden konnte, und versucht die Ausgabe mit dem nächsten Zeichen, und »C« bricht die Druckerausgabe ab und setzt das CTRL-P-Flag zurück. Weitere Versuche mit CTRL-P beachtet der Computer dann gar nicht mehr. Dieser Benutzerkomfort ist allerdings kein Verdienst von Digital Research, sondern von Amstrad, wo das BIOS entsprechend programmiert wurde.

(Martin Kotulla/hg)

Transient, resident oder was?

Nicht jeder Befehl unter CP/M steht permanent vollständig im Speicher. Der Komfort geht auf das Konto von Zusatzdateien.

Alle CP/M-Programme besitzen neben dem Dateinamen die Extension (Erweiterung) ».COM«. Besonders wichtige Befehle stehen aber ständig im RAM-Speicher – sind also »resident« (lat.: residere – verweilen). Dabei handelt es sich beispielsweise um Kommandos zum Anzeigen des Disketten-Inhaltsverzeichnisses oder zum Löschen von Dateien. Residente Kommandos werden blitzschnell ausgeführt. Man kann jedoch nicht alle Befehle im RAM aufnehmen; deshalb stehen die weiteren Kommandos auf der Diskette. Diese COM-Dateien heißen »transient« (lat.: transire – vorübergehen), weil sie nur vorübergehend in den Speicher geladen werden.

CP/M Plus kennt eine neue Befehlsklasse. Man könnte sie »halb-resident« oder »transient-resident« nennen. Gibt der Benutzer bei einem residenten Befehl Parameter an, die dieser normalerweise nicht kennt, so lädt der Computer selbsttätig eine Datei gleichen Namens in den Speicher. Diese erweiterten Anweisungen sind leistungsfähiger als die ursprünglichen residenten Befehle.

Beginnen wir mit den einfachen residenten Befehlen:

A: wählt das Laufwerk A als Standardlaufwerk.

B: selektiert das Diskettenlaufwerk B als angemeldete Station.

DIR zeigt das Inhaltsverzeichnis einer Diskette auf dem Bildschirm an. Wahlweise kann ein Bereich bestimmter Dateien mit den Wildcards (Joker-Zeichen) »?« und »*« ausgewählt werden. Der Computer zeigt dann nur die Dateinamen an, die ins vorgegebene Muster passen. Ein paar Beispiele für DIR finden Sie in Bild 1.

ERA oder **ERASE** löscht Dateien von der Diskette. Angegeben werden ein- oder mehrdeutige Dateinamen, sowie wahlweise das zugehörige Diskettenlaufwerk. Einige Beispiele für ERA finden Sie in Bild 2.

REN oder **RENAME** benennt Dateien auf der Diskette um. In der residenten Version des Befehls sind nur eindeutige Dateinamen gestattet. Als erstes muß der neue Dateiname eingetippt werden, dann der alte. Beide sind durch ein Gleichheitszeichen, ein Komma, ein Leerzeichen oder einen Tabulatorsprung voneinander getrennt:

```
A>REN NEU.TXT=ALT.TXT
A>RENAME B:NEUER.TXT=ALTER.TXT
A>REN A,B
A>REN B A
A>RENAME A<CTRL-I>B
```

Die Erweiterung erlaubt auch das gleichzeitige Umbenennen mehrerer Dateien:

```
A>REN *.TXT=*.BAK
A>RENAME DATEI.*=*.??1
A>REN *.BAK=*.*
```

TYP oder **TYPE** gibt eine ASCII-Datei auf dem Bildschirm aus. Sie läßt sich natürlich mit CTRL-P zusätzlich auf dem Drucker auflisten. Die Anzeige von COM-Files und anderen Nicht-Text-Dateien führt meist zu den tollsten Resultaten wie blinkenden Farben, Setzen von Windows und ähnlichen überraschenden Dingen. Das liegt daran, daß manche Codes mit TYPE als Bild-

schirmsteuerzeichen interpretiert werden. Sie sollten daher solche Dateien besser nicht ausdrucken. Beispiele für TYPE:

```
A>TYPE DATEI.TXT
A>TYP DATEI.TXT
```

Die Bildschirmausgabe können Sie mit CTRL-S anhalten und mit CTRL-Q wieder fortsetzen. Abgebrochen wird die Anzeige durch CTRL-C. Jedesmal, wenn der Bildschirm vollgeschrieben ist, erscheint die Systemmeldung

```
Press RETURN to Continue
```

und der Computer wartet auf einen Tastendruck. Dies muß übrigens nicht RETURN sein, jede beliebige andere Taste eignet sich dazu genauso. Dann erscheint allerdings das betreffende Zeichen auf dem Bildschirm.

USER oder **USE** wechselt den Benutzerbereich. Normalerweise ist die Benutzernummer 0 vorgegeben.

»USER 5« bedeutet zum Beispiel, daß DIR nur diejenigen Dateien auflistet, die im Benutzerbereich 5 gespeichert sind. Die Dateien aus den anderen Bereichen werden nicht angezeigt. Die Befehlssyntax sieht so aus:

```
USER 4
USE 0
USER 15
```

Unter CP/M Plus stehen dem Benutzer die Bereiche 0 bis 15 zur Verfügung. Ist ein anderer als der User-Bereich 0 gewählt, erscheint die Nummer im CP/M-Prompt:

```
A>USER 11
11A>USER 3
3A>USER 0
A>
```

Geben Sie nur »USER« ein, so fragt Sie der Computer nach der gewünschten Benutzernummer:

```
A>USER
Enter User #: 4
4A>
```

Es gibt noch einen einfacheren Weg, zwischen den verschiedenen Benutzerbereichen hin- und herzuschalten. Diese Methode ähnelt der Umschaltung zwischen einzelnen Laufwerken:

```
A>15A:
15A>7B:
7B>0:
B>A:
A>
```

Der Benutzerbereich 0 unterscheidet sich von allen anderen dadurch, daß Dateien mit SYS-Attribut (siehe Artikel über SET.COM in dieser Ausgabe), die in diesem Benutzerbereich stehen, von allen anderen Bereichen ebenfalls aufgerufen werden können. Der Benutzer 0 ist sozusagen ein »Super-User« oder »Super-Visor«. Den anderen Benutzern ist es nur gestattet, Daten vom Benutzerbereich 0 zu lesen und zu laden, nicht aber in diesen Bereich zu schreiben.

A>DIR	-alle Dateien der angemeldeten Diskettenstation werden ausgegeben.
A>DIR B:	-alle Dateien von Laufwerk B werden ausgegeben.
A>DIR *.COM	-alle Dateien mit dem Zusatz ».COM« werden angezeigt.
A>DIR KEYS.*	-alle Dateien mit dem Namen »KEYS« werden angezeigt.
A>DIR FILES.COM	-die Datei »FILES.COM« wird angezeigt.
A>DIR B:??BBC.*	-alle fünfbuchstabigen Dateien auf Laufwerk B, die »BBC« als dritten bis fünften Buchstaben haben, werden angezeigt.

Bild 1. Das Directory in allen Variationen

A>ERA FILE.COM	-die Datei »FILE.COM« wird gelöscht.
A>ERA *.COM	-alle Dateien mit dem Zusatz ».COM« werden gelöscht.
A>ERASE *.COM	-alle Dateien mit dem Zusatz ».COM« werden gelöscht.
A>ERA A?CDEF.??T	-alle sechsbuchstabigen Dateien mit einem »A« als ersten und »CDEF« als dritten bis sechsten Buchstaben, sowie einem Zusatz mit »T« als letzten Buchstaben, werden gelöscht.
A>ERA *.*	-alle Dateien werden gelöscht.
A>ERASE *.*	-alle Dateien werden gelöscht.

Bild 2. Löschen mit Komfort

```
Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

    Name       Bytes   Recs   Attributes       Name       Bytes   Recs   Attributes
------------ ------ ------ ------------   ------------ ------ ------ ------------
AMSDOS   COM     1k      8 Dir RW         BANKMAN  BAS     1k      7 Dir RW
BANKMAN  BIN     2k     12 Dir RW         C10CPM3  EMS    25k    200 Dir RW
DATE     COM     3k     23 Dir RW         DEVICE   COM     8k     58 Dir RW
DIR      COM    15k    114 Dir RW         DISCKIT3 COM     6k     48 Dir RW
ED       COM    10k     73 Dir RW         ERASE    COM     4k     29 Dir RW
GET      COM     7k     51 Dir RW         KEYS     CCP     1k      3 Dir RW
KEYS     WP      1k      3 Dir RW         LANGUAGE COM     1k      8 Dir RW
PALETTE  COM     1k      8 Dir RW         PIP      COM     9k     68 Dir RW
PROFILE  SUB     1k      1 Dir RW         PUT      COM     7k     55 Dir RW
RENAME   COM     3k     23 Dir RW         SET      COM    11k     81 Dir RW
SET24X80 COM     1k      8 Dir RW         SETDEF   COM     4k     32 Dir RW
SETKEYS  COM     2k     16 Dir RW         SETLST   COM     2k     16 Dir RW
SETSIO   COM     2k     16 Dir RW         SHOW     COM     9k     66 Dir RW
SUBMIT   COM     6k     42 Sys RW         TYPE     COM     3k     24 Dir RW

Total Bytes     =    146k  Total Records =     1093  Files Found =   28
Total 1k Blocks =    146   Used/Max Dir Entries For Drive A:    29/  64
```

Bild 3. Ganz schön leistungsfähig, dieses DIR[FULL]

Nebenbei erwähnt: In CP/M Plus dürfen Sie mehrere Befehle in einer einzigen Zeile eingeben. Sie werden durch Ausrufezeichen voneinander getrennt:

```
A>DIR ! ERA FILE.COM ! DIR
```

Wenn Sie nun auf die Erläuterung des unter CP/M 2.2 residenten SAVE-Befehls warten, müssen wir Sie enttäuschen. Dieses Kommando, das Teile der TPA auf der Diskette speichert, ist unter CP/M Plus nicht mehr resident, sondern steht als SAVE.COM auf der Diskette. Das erschwert die Arbeit oft unnötig, weshalb diese Maßnahme wohl seitens der Benutzer keine Begeisterungsstürme auslöst.

Sozusagen als Ersatz versteht der Computer jetzt das Kommando **DIRSYS**. Dieser Befehl zeigt Systemdateien an einer Diskette an. Er hat die gleiche Syntax wie DIR und kann auch zu **DIRS** abgekürzt werden:

```
A>DIRSYS
A>DIRS
A>DIRSYS.COM
A>DIRS ??AAB.COM
```

Gehen wir nun zu den »Mischbefehlen« über. Von einem schönen alphabetisch sortierten Directory samt Angabe der Dateigrößen und anderer Informationen träumt wohl jeder.

DIR.COM unter CP/M Plus kann das auch. Sie erinnern sich: Schwierige Aufträge kann ein residenter Befehl nicht bearbeiten und reicht sie an das transiente Programm weiter. DIR.COM (die Erweiterung von DIR) steht beim 6128 auf der ersten CP/M-Systemdiskette. Geben Sie

```
A>DIR [FULL]
```

ein und es erscheint die Meldung

```
Scanning Directory...
Sorting  Directory...
```

Der Computer gibt dann sortiert den Disketteninhalt aus und zwar so, wie es Bild 3 zeigt.

Weitere nützliche Fähigkeiten von DIR.COM zeigen[DIR], das ausschließlich Nicht-Systemdateien auflistet, [SYS], das nur die Systemfiles ausgibt und [NOPAGE], das die Aufforderung »Press RETURN to Continue« bei vollgeschriebenem Bildschirm unterdrückt.

Die Anweisung [NOSORT] unterläßt die alphabetische Sortierung der Dateinamen. [RO] zeigt nur schreibgeschützte Files (Read/Only Files) an,

ATT	–Ausgabe der benutzerdefinierten Dateiattribute F1, F2, F3 und F4.
DATE	–Zusätzliche Ausgabe der Datums- und Uhrzeit. Sind die Disketten nicht mit INITDIR.COM vorbereitet, erscheint die Fehlermeldung »ERROR: Date and Time Stamping Inactive«.
DIR	–Es werden nur diejenigen Dateien gezeigt, die keine Systemfiles sind.
DRIVE=d	–Das Directory wird vom angegebenen Laufwerk geholt.
DRIVE=(d1,d2,...)	–Das Inhaltsverzeichnis der angegebenen Laufwerke wird angezeigt.
DRIVE=ALL	–Das Inhaltsverzeichnis aller angeschlossenen Laufwerke wird ausgegeben.
EXCLUDE	–Alle angegebenen Dateinamen werden bei der Ausgabe des Inhaltsverzeichnisses von DIR.COM unterdrückt.
FF	–Ist der Drucker mit CTRL-P zugeschaltet, wird vor der Ausgabe des Directory ein Line-Feed gesendet. Über LENGTH läßt sich die Seitenlänge festlegen.
FULL	–DIR.COM gibt ein vollständiges Inhaltsverzeichnis aus.
Gn	–Zeigt das Inhaltsverzeichnis des angegebenen Benutzerbereichs.
LENGTH=n	–Bestimmt die Seitenlänge, also die Zahl der Zeilen, nach der jeweils ein Form-Feed an den Drucker geschickt wird.
MESSAGE	–Sagt dem DIR-Programm, daß alle durchsuchten Laufwerke und Benutzerbereiche angezeigt werden sollen. »DIR[USER=ALL,MESSAGE]« gibt in allen Benutzerbereichen, in denen keine Dateien gefunden werden, »Directory For Drive d: User n – No File« aus.
NOPAGE	–Unterdrückt die Ausgabe von »Press RETURN to Continue« am Seitenende.
NOSORT	–Verhindert die alphabetische Sortierung der Dateinamen.
RO	–Zeigt nur Dateien mit dem Read/Only-Attribut.
RW	–Zeigt nur Dateien mit dem Read/Write-Attribut.
SIZE	–Gibt ein vereinfachtes Directory-Listing aus.
SYS	–Zeigt nur Dateien mit dem SYS-Attribut.
USER=n	–Listet die Dateien eines speziellen Benutzerbereichs.
USER=(0,1,...)	–Gibt auf dem Bildschirm die Dateien aller angegebenen Benutzerbereiche aus.
USER=ALL	–Zeigt das Directory aller Benutzerbereiche.

Bild 4. Alle Befehlsformen von DIR.COM auf einen Blick

[RW] hingegen nur solche ohne Schreibschutz (Read/Write Files). [DATE] gibt die Datums- und Uhrzeitkennung der Dateien mit aus, wenn die Diskette mit INITDIR.COM (siehe Artikel »Sekundengenau« in dieser Ausgabe) entsprechend vorbereitet wurde. [ATT] zeigt die vom Benutzer definierbaren Dateiattribute F1, F2, F3 und F4 zusätzlich mit an – sofern sie eingeschaltet sind.

Statt der Angabe »DIR B:[Option]« können Sie aber auch »DIR [DRIVE=B,Option]« eingeben. Hier erkennen Sie, wie Sie mehrere Optionen gleichzeitig spezifizieren. Diese werden durch Kommas getrennt.

```
DIR [DRIVE=A,FULL,NOPAGE,DATE]
```

Doch DIR.COM kann noch mehr. Statt »DRIVE=d« versteht es auch »DRIVE=ALL« und gibt dann das Directory aller angeschlossenen Diskettenstationen aus. Interessiert der Disketteninhalt von mehreren der angeschlossenen Laufwerke, verwendet man den Befehl »[DRIVE=(A,B,...)]«.

[SIZE] sortiert die Dateinamen nicht nach dem Alphabet, sondern nach der Dateigröße.

Um die Dateien eines speziellen Benutzerbereichs aufzulisten, eignen sich zwei Kommandos. Das eine lehnt sich an den residenten USER-Befehl an, das andere an das Programm PIP.COM. [USER=n] entspricht [Gn].

```
A>DIR [USER=5]
A>DIR [G5]
```

Der G-Befehl wurde allerdings von Digital Research offiziell nicht dokumentiert und entsprang wohl einer Laune des Programmierers.

[USER=n] hat eine Zusatzoption – und zwar ALL.[USER=ALL].

```
A>DIR [FULL,USER=ALL]
```

Ebenso können mehrere Benutzerbereiche gleichzeitig gelistet werden.

```
A>DIR [FULL,USER=(0,1,2,3,4)]
```

Die Option MESSAGE erlaubt es dem Benutzer, die Arbeit von DIR.COM zu verfolgen. Das Programm zeigt dann ständig an, welches Laufwerk und welche Benutzer-Nummer gerade untersucht wird.

Speziell für die Druckerausgabe sind die Anweisungen FF und LENGTH gedacht. FF weist DIR.COM an, vor der Ausgabe einen Blattvorschub (Form-Feed) an den Drucker zu senden. LENGTH=n setzt die Seitenlänge in Druckzeilen fest.

Zuletzt noch die EXCLUDE-Option. Wie der Name schon andeutet, lassen sich bestimmte Dateien aus dem Inhaltsverzeichnis wieder herausnehmen. So druckt

```
A>DIR [EXCLUDE] *.COM
```

alle Dateien bis auf die COM-Files aus.

Eine vollständige Darstellung aller DIR-Optionen zeigt Bild 4.

Erheblich komfortabler als unter CP/M 2.2 arbeitet auch ERA.COM. Fragt die Version 2.2 nur nach der Eingabe von »ERA *.*«, ob das Löschen der gesamten Diskette wirklich im Sinne des Anwenders ist, geschieht das unter CP/M Plus bei jedem mehrdeutigen Dateinamen, in dem ein Fragezeichen oder Stern vorkommt.

```
A>ERA *.*
ERASE *.* (Y/N)?
A>ERA F.*
ERASE F.* (Y/N)?
A>ERA R??.COM
ERASE R??.COM (Y/N)?
```

Die eigentliche Datei ERA.COM wird erst aktiv, wenn Sie »ERA« ohne Parameter eingeben. Das Programm fragt dann nach den zu löschenden Files:

```
A>ERA
Enter filename: DATEI
A: DATEI   .    (Y/N)? Y
```

Bei mehrdeutigen Dateinamen zeigt ERA.COM alle in Frage kommenden Files und gibt dem Benutzer Gelegenheit anzugeben, ob die Dateien gelöscht werden sollen oder nicht.

```
A>ERA
Enter filename: DATEI.*
A: DATEI   .    (Y/N)? Y
A: DATEI   .2   (Y/N)? N
```

Die Abfrage von mehrdeutigen Dateinamen kann auch bei der direkten Angabe der ERA-Parameter über »CONFIRM« erfolgen.

```
A>ERA *.COM [CONFIRM]
A: SUBMIT  .COM (Y/N)? N
A: SETKEYS .COM (Y/N)? Y
A: LANGUAGE.COM (Y/N)? ^C
*** Aborted by ^C ***
```

Der normale RENAME-Befehl gestattet weder unter CP/M 2.2 noch unter 3.0 die Angabe mehrdeutiger Dateinamen. RENAME.COM nun macht's möglich.

```
A>REN
Enter New Name: *.CIM
Enter Old Name: *.COM
```

War CP/M 2.2 noch darauf ausgerichtet, daß eine eventuelle Laufwerksbezeichnung nur beim ersten oder bei beiden Dateinamen (aber dann identisch) angegeben werden durfte, ist es der Version 3.0 »egal«, ob Sie den Laufwerksnamen beim ersten oder zweiten Dateinamen nennen.

Ein ohne Parameter aufgerufenes »TYPE« veranlaßt den Computer, »TYPE.COM« in den Speicher zu laden und die Anfrage »Enter file:« auszugeben:

```
A>TYPE
Enter file:
```

Gefällt Ihnen nicht, nach jeweils 24 oder 25 ausgegebenen Zeilen eine Taste drücken zu müssen (»Press RETURN to Continue«), können Sie diese Funktion mit [NOPAGE] abschalten:

```
A>TYPE TEXT.DAT[NOPAGE]
```

Auch die Schreibweise »NO PAGE«, also mit einem eingefügten Leerzeichen, ist dem System verständlich.

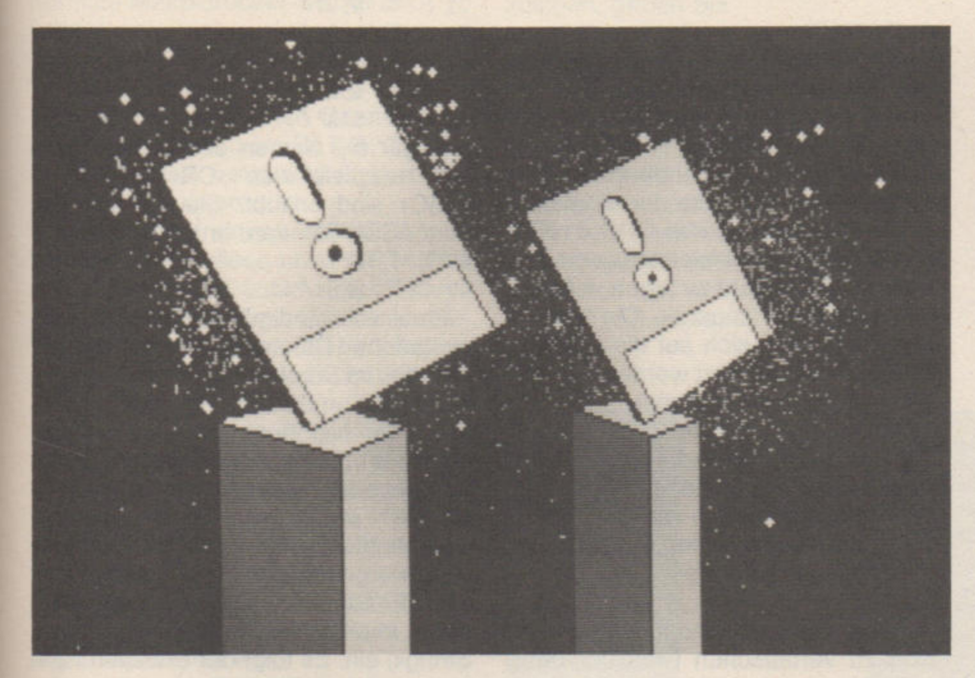
```
A>TYPE TEXT.DAT[NO PAGE]
```

Die Option »PAGE« ist ebenso zulässig, kann aber wegfallen, weil sie schon standardgemäß voreingestellt ist.

(Martin Kotulla/hg)



Spezialitäten



Der Schneider CPC 6128 wird mit verschiedenen, speziell für ihn geschriebenen Programmen ausgeliefert. Was diese alles leisten, das finden Sie nicht in Ihrem Handbuch.

Neben den eigentlichen Systemdateien enthält die CP/M-Diskette des Schneider CPC 6128 verschiedene Hilfsprogramme, die die Arbeit mit dem Computer komfortabler machen. Was fehlt, ist eine ausführliche Beschreibung im Handbuch.

AMSDOS.COM ist eine sehr kurze Datei (1 KByte), die die Rückkehr aus CP/M ins Basic ermöglicht. Dazu ruft das Programm den RSX-Befehl »IBASIC« auf. Da dabei alle Daten im Speicher verlorengehen, kann man auch einen einfacheren Weg wählen, indem man einfach die Tasten CTRL, SHIFT und ESC gleichzeitig drückt.

Bei **DISCKIT3.COM** handelt es sich um ein sehr benutzerfreundliches Programm, mit dem sich Disketten kopieren, formatieren und verifizieren (die Kopie auf Fehler überprüfen) lassen. Das Programm ermittelt automatisch die Zahl der angeschlossenen Diskettenlaufwerke. Es wird mit den Funktionstasten gesteuert und ist nahezu narrensicher zu bedienen. Diese Sicherheit erfordert allerdings auch etwas Aufwand. So muß der Benutzer etwa jede Auswahl, die er getroffen hat, bestätigen, und beim Kopieren von Disketten diese zwingend aus dem Laufwerk entfernen, um anschließend eine Taste drücken zu können. Das Programm überprüft sogar, ob das Laufwerk dann auch wirklich leer ist.

Das Hauptmenü von DISCKIT3 stellt die Wahl frei zwischen »Copy« (sektorenweises Kopieren von Disketten), »Format« (Formatieren der Disketten), »Verify« (Überprüfung von Disketten) sowie »Exit from program«, das die Bearbeitung des Programms abbricht.

LANGUAGE.COM schaltet zwischen verschiedenen nationalen Zeichensätzen um. Das betrifft aber leider nur die Bildschirmausgabe und nicht die Tastaturbelegung. Letztere erfordert die Umstellung mit **SETKEYS.COM**. Language benutzt folgende Parameter:

```
A>LANGUAGE 0 -
US-amerikanische Zeichen
A>LANGUAGE 1 -
Französische Zeichen
A>LANGUAGE 2 -
Deutsche Zeichen
A>LANGUAGE 3 -
Britische Zeichen
A>LANGUAGE 4 -
Dänische Zeichen
A>LANGUAGE 5 -
Schwedische Zeichen
A>LANGUAGE 6 -
Italienische Zeichen
A>LANGUAGE 7 -
Spanische Zeichen
```

Bei dieser Gelegenheit wollen wir auch darauf hinweisen, daß CP/M 3.0 auf dem Schneider einen völlig anderen Zeichensatz benutzt als CP/M 2.2. Das betrifft vor allem die Grafikzeichen, die vollständig durch andere Symbole und eine Reihe nationaler Zeichen mit den verschiedensten Akzenten und Tilden ersetzt wurden. Aber auch bei den Buchstaben »O«, »Q« und der »0« (Null) änderte sich die Definition. Aufgrund dieser Änderungen sind viele professionelle Anwenderprogramme befähigt, ohne Schwierigkeiten auf die verschiedensten Zeichensätze zuzugreifen. Allerdings läuft CP/M 2.2-Software nicht mehr korrekt, wenn sie auf Grafikzeichen angewiesen ist.

Das Programm **PALETTE.COM** paßt Bildschirm- und Zeichenfarben an. Der Aufruf erfolgt mit

```
A>PALETTE Schirm-, Schriftfarbe
```

Für schwarze Schrift auf leuchtend weißem Grund mit leuchtend weißem Bildschirmrand geben Sie beispielsweise

```
A>PALETTE 63,0
```

ein. Auch die hexadezimale Darstellung der Zahlen für die Farben ist zulässig:

```
A>PALETTE &3F,&00
```

Eine getrennte Festlegung der Hintergrund- und der Randfarbe – sowie Blinken – läßt sich mit PALETTE nicht einstellen.

Wenn Sie unter CP/M im Modus 1 oder 0 arbeiten wollen, können Sie weitere Farben festlegen. Dazu fügen Sie bei PALETTE die Werte für die Farbregister 0 bis 15 hinzu. Allerdings verläuft dann der Bildschirmaufbau vieler Programme, die ja meist mit 80 Zeichen pro Zeile arbeiten, nicht korrekt.

```
A>PALETTE Farbe0 Farbe1 Farbe2
Farbe3 Farbe4 Farbe5 Farbe6
Farbe7 Farbe8 Farbe9 Farbe10
Farbe11 Farbe12 Farbe13 Farbe14
Farbe15
```

Natürlich dürfen Sie auch einige der Register frei lassen. Um beispielsweise nur fünf Farben (0 bis 4) zu definieren, tippen Sie:

```
A>PALETTE 0 1 0 2 3
```

Unter Basic ginge das mit:

```
10 INK 0,0
20 INK 1,1
30 INK 2,0
40 INK 3,2
50 INK 4,3
```

Die Farbnummern entsprechen nicht denen des Basic-Interpreters, sondern sind sehr trickreich codiert. Alle Werte dürfen im Bereich zwischen 0 und 63 liegen. Größere Werte als 27 werden logisch verknüpft, bis ein passender Wert herauskommt.

Die Zahlen von 0 bis 63 lassen sich in sechs Bit darstellen. Je zwei Bit der Zahl geben die Intensität einer der drei Grundfarben Rot, Grün und Blau an. Bit 0 und 1 spezifizieren die Intensität des Blau-Anteils, Bit 2 und 3 des roten und Bit 4 und 5 des grünen Anteils.

Mit zwei Bit lassen sich 2^2, also vier verschiedene Intensitäten darstellen. In allen Kombinationen ergibt das mehr



Farben, als der Video-Chip wiedergeben kann. Deshalb werden die Intensitäten intern auf drei verschiedene Werte beschränkt.

- Intensität 00 bin (0 dez) entspricht Intensität 0.
- Intensität 01 bin (1 dez) entspricht Intensität 1.
- Intensität 10 bin (2 dez) entspricht Intensität 1.
- Intensität 11 bin (3 dez) entspricht Intensität 2.

Somit ist es ohne Bedeutung, ob Sie als Intensität 01 oder 10 bin angeben.

Die Farbwerte für PALETTE.COM im Vergleich mit den Werten für den Basic-Befehl INK finden Sie in Bild 1.

Durch den Aufruf von **SET24X80.COM** kann man zwischen 24 und 25 Bildschirmzeilen hin- und herschalten. Das ist deshalb notwendig, weil der Schneider-Computer mit 25 Zeilen pro Bildschirmseite, einige CP/M-Programme aber nur mit einer 24zeiligen Darstellung korrekt arbeiten.

```
A>SET24X80 ON
A>SET24X80 OFF
```

Denn bei einigen Versionen des Schneider CPC 6128 läuft dieses Programm nicht perfekt. Also kann leider nicht jeder Schneider-Fan CP/M auf 24 Zeilen verteilt bestaunen.

Tastatur mit Komfort

SETKEYS.COM definiert die Tastatur um. Die Routine entspricht damit den Basic-Befehlen KEY und KEY DEF. Auf der Systemdiskette 1 befinden sich zwei bereits für bestimmte Zwecke vorbereitete Dateien – nämlich **KEYS.CCP** und **KEYS.WP**. Die erste Datei paßt die Sondertasten des Computers zur Arbeit im CCP (Console Command Processor) an. Die KEYS.WP-Datei eignet sich zur Benutzung von Textverarbeitungs-Programmen. »WP« steht dabei für »Word-Processor«:

```
A>SETKEYS KEYS.CCP
A>SETKEYS KEYS.WP
```

Auf der dritten Seite der Systemdisketten stoßen Sie noch auf **KEYS.DRL**, eine spezielle Tastaturbelegung für Dr Logo.

```
A>SETKEYS KEYS.DRL
A>LOGO3
```

Um eine solche Datei selbst zusammenzubauen, benötigen Sie einen Texteditor, beispielsweise Wordstar im N-Modus (»Non-Document« – Bearbeiten einer Programmdatei) oder auch »PIP KEYS.DAT=CON:«. PIP erfordert am Zeilenende allerdings ein RETURN und CTRL-J, am Ende der Datei dann CTRL-Z. Tippfehler dürfen Sie sich dabei nicht erlauben. Ansonsten eignet sich jeder Editor, der reine ASCII-Files erzeugt.

```
10 OPENOUT "KEYS.DAT"
20 PRINT #9
.
.
.
90 CLOSEOUT
```

In der neuen Datei definiert jede Zeile eine Taste um. Dazu brauchen Sie den Code der Taste. Das ist der gleiche, der auch für den Befehl KEY DEF benutzt wird. Dieser ist im Handbuch – aber auch auf der Oberseite der eingebauten Diskettenstation des CPC 6128 – abgedruckt. Um beispielsweise ein Zeichen auf die Z-Taste zu legen, müssen Sie den Code 71 angeben. Die nächste Eingabe bezieht sich auf die Tastaturebene, die umdefiniert werden soll. »N« steht dabei für »Normal«, »S« für »SHIFT« und »C« für die CTRL-Ebene.

Die deutsche DIN-Tastatur hat gegenüber der amerikanischen Norm »Z« und »Y« vertauscht. Wollen wir nun das »Y« auf die Normalebene der Z-Taste legen, so müssen wir eingeben:

```
71 N "y"
```

Um alle drei Ebenen der Z- und der Y-Taste zu vertauschen (was der deutschen Norm entspricht), schreiben wir folgende Befehle in die Datei:

```
71 N "y"
71 S "Y"
71 C "^Y"
43 N "z"
43 S "Z"
43 C "^Z"
```

Control-Codes, deren ASCII-Wert zwischen 0 und 31 liegt, lassen sich also durch einen vorhergehenden Pfeil nach oben eingeben. Die Beschreibung der Control-Funktionen finden Sie in dem Artikel »CP/M mit Nachbrenner« in diesem Heft.

Soll auf alle drei Tastenebenen derselbe Code gelegt werden, können Sie sich etwas Arbeit sparen:

```
71 N S C " "
```

Die Z-Taste ist jetzt in allen drei Ebenen mit dem Leerzeichen belegt. Statt des gewünschten Zeichens selbst darf auch der entsprechende ASCII-Wert eingesetzt werden. Dieser wird mit »^'« eingeleitet und mit »'« abgeschlossen.

Die Tabelle

```
71 N "^'121'"
71 S "^'89'"
71 C "^'25'"
43 N "^'122'"
43 S "^'90'"
43 C "^'26'"
```

vertauscht auch die Z- mit der Y-Taste.

Hexadezimalen Zahlen stellt man ein Doppelkreuz voran.

```
71 N "^'#79'"
71 S "^'#59'"
71 C "^'#19'"
43 N "^'#7A'"
43 S "^'#5A'"
43 C "^'#1A'"
```

Wahlweise wird auch das kaufmännische Und-Zeichen als Kennung für hexadezimale Werte akzeptiert.

```
71 N "^'&79'"
71 S "^'&59'"
71 C "^'&19'"
43 N "^'&7A'"
43 S "^'&5A'"
43 C "^'&1A'"
```

Sogar die Namen der Steuercodes wie beispielsweise »CR«, »FF« und »ESC« sind erlaubt. Sie müssen nur dem ASCII-Standard entsprechen.

```
47 S "^'FF'"
47 C "^'ESC'"
```

Eine vollständige Tabelle der verschiedenen Darstellungen für die Steuercodes im Bereich zwischen 0 und 31 finden Sie im Bild 2.

Bei der hier besprochenen Routine dürfen am Ende jeder Zeile Kommentare eingefügt werden.

```
71 N "^'#79'" verwende y statt z
```

Auch die Belegung der Funktionstasten geschieht mit Hilfe von SETKEYS.COM. Die Befehlszeilen dafür leitet ein »E« (für »Expansion String«) ein. Es folgt der Erweiterungscode und dann der Funktionsstring. Um beispielsweise die Taste F0 mit DIR zu belegen, schreibt man

```
E 128 "DIR^M"
```

Die Angabe »^M« sorgt dafür, daß ein Carriage-Return mit in den String aufgenommen wird, so daß sich nach Betätigen der Funktionstaste F0 die RETURN-Taste erübrigt.

Die Nummer des Erweiterungscodes darf wieder in hexadezimaler Schreibweise eingefügt werden – auch hier mit dem kaufmännischen Und-Symbol »&« oder dem Nummernzeichen »#«:

```
E &80 "DIR^M"
E #81 "AMSDOS^M"
```

Druckersteuerung mit Leichtigkeit

Eine ähnliche Datei wie SETKEYS.COM verwendet auch **SETLST.COM**. Dieses CP/M-Programm versorgt den Drucker (»List Device«) mit Steuercodes. Wollen Sie zum Beispiel den NLQ401 von Schneider oder den baugleichen Drucker M-1009 von Brother auf Briefqualität (Near Letter Quality) umschalten, geben Sie unter Basic

```
PRINT #8,CHR$(27);CHR$(73);
CHR$(3)
```

ein. Unter CP/M Plus brauchen Sie dazu eine Datei mit folgendem Aussehen:

```
^'ESC'
^'73'
^'&03'
```

Wenn Sie diese Datei mit dem Namen »NLQ« speichern, dann können Sie

unter CP/M auf die NLQ-Briefqualität mit »SETLST NLQ« umschalten. Aber auch der umgekehrte Weg ist möglich. Die Datei NLQOFF schaltet wieder auf die normale Matrixdrucker-Schrift um.

```
^'ESC'
I
^A
```

Da die verschiedenen Darstellungsarten mit denen von SETKEYS.COM identisch sind, zeigt Bild 2 auch die erlaubten Druckercodes.

Wer Besitzer einer seriellen Schnittstelle ist, kann diese unter CP/M Plus mit **SETSIO.COM** ansteuern. Das von Schneider angebotene Interface arbeitet allerdings nicht mit SETSIO zusammen. Sie müssen auf die RS232-Erweiterung von Amstrad zurückgreifen. Deshalb nämlich, da Schneider seine Eigenentwicklung zu einem möglichst günstigen Preis unter die Leute bringen wollte. So baute man in das Interface einen Z80-STI-Chip (»Serial Timer and Interface«) ein, anstelle der ICs Z80-SIO (»Serial Input/Output«) und Intel-8255. Das ist zwar billiger, aber die Kompatibilität fehlt.

Der einfache SETSIO-Befehl testet, ob überhaupt eine serielle Schnittstelle angeschlossen ist:

```
A>SETSIO
SIO not found
```

Diverse Parameter stellen die Schnittstelle ein. So legt RX die Übertragungsgeschwindigkeit in Baud für den Empfang (TX für das Senden) fest. »BITS n« bestimmt die Zahl der Datenbit und »STOP n« die der Stopbit.

Um beispielsweise die Geschwindigkeit beim Empfang auf 4800 Baud festzulegen, geben Sie

```
A>SETSIO RX 4800
```

ein. Soll die Geschwindigkeit für Senden und Empfangen den gleichen Wert aufweisen, so bewirkt das

```
A>SETSIO RX 9600, TX 9600
```

Einfacher wird es mit

```
A>SETSIO 9600
```

Mehrere Parameter dürfen gleichzeitig angegeben werden, müssen aber durch Kommas getrennt sein.

```
A>SETSIO RX 19200, STOP 2,
BITS 6
```

Die komplette Liste aller Anweisungen finden Sie in Bild 3.

Damit haben wir Ihnen alle Programme vorgestellt, die es nur bei Schneider-Computern unter CP/M Plus gibt und die gleichzeitig unmittelbar mit dem Betriebssystem zusammenhängen. Nicht direkt zu CP/M Plus gehört der Logo-Interpreter, der auf der dritten Seite der Systemdisketten in den Dateien LOGO3.COM und LOGO3.SUB enthalten ist. Eine Einführung in Logo finden Sie im 2. Schneider-Sonderheft (Sonderheft 1/86 von Happy-Computer).

(Martin Kotulla/hg)

Farbnummer unter CP/M		Farbbezeichnung	Farbnummer unter Basic
0	00 hex	Schwarz	0
2	02 hex	Blau	1
3	03 hex	Hellblau	2
8	08 hex	Rot	3
10	0A hex	Magenta	4
11	0B hex	Hellviolett	5
12	0C hex	Hellrot	6
14	0E hex	Purpur	7
15	0F hex	Helles Magenta	8
32	20 hex	Grün	9
34	22 hex	Blaugrün	10
35	23 hex	Himmelblau	11
40	28 hex	Gelb	12
42	2A hex	Weiß	13
43	2B hex	Pastellblau	14
44	2C hex	Orange	15
46	2E hex	Rosa	16
47	2F hex	Pastellmagenta	17
48	30 hex	Hellgrün	18
50	32 hex	Seegrün	19
51	33 hex	Helles Blaugrün	20
56	38 hex	Limonengrün	21
58	3A hex	Pastellgrün	22
59	3B hex	Pastellblaugrün	23
60	3C hex	Hellgelb	24
62	3E hex	Pastellgelb	25
63	3F hex	Leuchtend weiß	26

Bild 1. Reiche Auswahl – alle Farben von PALETTE.COM

Dezimal	Hexadezimal (1)	Hexadezimal (2)	Controlcode	Codename
^'0'	^'&00'	^'#00'	^%	^'NUL'
^'1'	^'&01'	^'#01'	^A	^'SOH'
^'2'	^'&02'	^'#02'	^B	^'STX'
^'3'	^'&03'	^'#03'	^C	^'ETX'
^'4'	^'&04'	^'#04'	^D	^'EOT'
^'5'	^'&05'	^'#05'	^E	^'ENQ'
^'6'	^'&06'	^'#06'	^F	^'ACK'
^'7'	^'&07'	^'#07'	^G	^'BEL'
^'8'	^'&08'	^'#08'	^H	^'BS'
^'9'	^'&09'	^'#09'	^I	^'HT'
^'10'	^'&0A'	^'#0A'	^J	^'LF'
^'11'	^'&0B'	^'#0B'	^K	^'VT'
^'12'	^'&0C'	^'#0C'	^L	^'FF'
^'13'	^'&0D'	^'#0D'	^M	^'CR'
^'14'	^'&0E'	^'#0E'	^N	^'SO'
^'15'	^'&0F'	^'#0F'	^O	^'SI'
^'16'	^'&10'	^'#10'	^P	^'DLE'
^'17'	^'&11'	^'#11'	^Q	^'DC1'
^'18'	^'&12'	^'#12'	^R	^'DC2'
^'19'	^'&13'	^'#13'	^S	^'DC3'
^'20'	^'&14'	^'#14'	^T	^'DC4'
^'21'	^'&15'	^'#15'	^U	^'NAK'
^'22'	^'&16'	^'#16'	^V	^'SYN'
^'23'	^'&17'	^'#17'	^W	^'ETB'
^'24'	^'&18'	^'#18'	^X	^'CAN'
^'25'	^'&19'	^'#19'	^Y	^'EM'
^'26'	^'&1A'	^'#1A'	^Z	^'SUB'
^'27'	^'&1B'	^'#1B'	^[	^'ESC'
^'28'	^'&1C'	^'#1C'	^|	^'FS'
^'29'	^'&1D'	^'#1D'	^]	^'GS'
^'30'	^'&1E'	^'#1E'	^^	^'RS'
^'31'	^'&1F'	^'#1F'	^_	^'US'

Bild 2. Die Steuercodes für SETKEYS.COM und SETLST.COM

RX n	– Baudrate der Empfangsgeschwindigkeit. »n« muß zwischen 50 und 19200 liegen.
TX n	– Baudrate der Sendegeschwindigkeit. »n« muß zwischen 50 und 19200 liegen.
n	– Baudrate der Sende- und Empfangsgeschwindigkeit. »n« muß zwischen 50 und 19200 liegen.
BITS n	– Zahl der Datenbits bei der Übertragung. »n« ist zwischen 5 und 8 zulässig.
STOP n	– Zahl der Stopbits bei der Übertragung. »n« darf die Werte 1, und 2 annehmen.
PARITY	– Bestimmt die Parität. Erlaubt ist PARITY EVEN, PARITY ODD und PARITY NONE.
XON	– Legt das XON-Übertragungsprotokoll fest. Zulässig sind XON ON und XON OFF.
HANDSHAKE	– Übertragung über Handshake ein- beziehungsweise ausschalten. Möglich sind HANDSHAKE ON und HANDSHAKE OFF.

Bild 3. Freie Auswahl – die Optionen von SETSIO.COM

PIP.COM kopiert alles

Eines der wichtigsten System-Programme unter CP/M ist PIP.COM. Es überträgt Daten und Dateien zwischen allen Peripheriegeräten, die das Betriebssystem ansprechen kann.

PIP.COM ist die Abkürzung für »Peripheral Interchange Program« – auf Deutsch »Programm zum Austausch von Daten zwischen Peripheriegeräten«. Um das Betriebssystem zu verstehen, muß man zwei Gruppen von Peripheriegeräten unterscheiden, die unter CP/M grundsätzlich verschieden behandelt werden. Zu der einen Gruppe zählen Disketten- und Festplattendateien – in der anderen sind die zeichenorientierten Geräte wie der Bildschirm, die Tastatur und der Drucker zu Hause.

PIP.COM erlaubt das Kopieren von Diskettendateien mit folgendem Befehl:

```
A>PIP KOPIE.DAT=ORIGINAL.DAT
```

Allgemeiner läßt sich die Anweisung mit

```
A>PIP d:ziel.typ=d:quelle.typ
```

ausdrücken. Soll die Zieldatei denselben Namen tragen wie die Quelldatei, so erspart man sich etwas Arbeit, indem man den Zielnamen wegläßt und nur die Laufwerksbezeichnung angibt.

```
A>PIP d:=d:quelle.typ
```

Ein Beispiel:

```
A>PIP B:=A:DATEI
```

Stimmt die Laufwerksbezeichnung in einem Dateinamen mit der im CP/M-Prompt überein – steht die Datei also auf der Diskette in dem gerade aktiven Laufwerk –, kann sie wegfallen.

```
A>PIP B:=DATEI
```

Gegenüber PIP.COM unter CP/M 2.2 dürfen Sie bei CP/M 3.0 die Reihenfolge – ähnlich wie bei RENAME – verdrehen. Völlig gleichwertig mit dem gezeigten Aufruf von PIP.COM steht

```
A>PIP B:DATEI=A:
```

In den meisten Fällen wird PIP.COM beim Kopieren einer Datei von einer Diskette auf eine andere benötigt. Dafür braucht man aber zwei Diskettenlaufwerke, wenn man sich nicht die Fähigkeit des CPC 6128 zunutze macht, das zweite Laufwerk zu simulieren.

PIP.COM kann auch mehrere Dateien auf einmal kopieren. Dies geschieht durch Angabe mehrdeutiger Dateinamen. Um etwa den gesamten Disketteninhalt von Laufwerk A auf die Diskette in Laufwerk B zu übertragen, benutzen Sie

```
A>PIP B:=A:*.*
```

Will man ausschließlich COM-Files kopieren, deren erster Buchstabe ein »A« ist, empfiehlt sich unter PIP folgender Aufruf:

```
A>PIP B:=A:A*.COM
```

Mit den Wildcards »*« und »?« sind die verschiedensten Anweisungen möglich.

Oft möchte man bestimmte Programme in einen anderen Benutzerbereich kopieren. Dazu müssen Sie bei der Zieldatei den gewünschten Benutzerbereich zusammen mit einem »G« in eckigen Klammern hinter dem Namen angeben.

```
A>PIP BENUTZ5[G5]=BENUTZER.0
```

Lassen Sie die G-Option (»Go to user area«) weg, kopiert PIP.COM immer nur Dateien im aktuellen Benutzerbereich.

Die G-Option kann aber auch bei der Quelldatei hinzugefügt werden. Dort bestimmt sie, aus welchem Benutzerbereich sie geholt werden soll (»Get from User«). Eine Datei »BENUTZER.9« aus dem Bereich 9 kann so beispielsweise nach 0 (den aktiven User-Bereich) als »BENUTZER.0« übertragen werden:

```
A>PIP BENUTZER.0=BENUTZER.10[G9]
```

Aber auch zwischen verschiedenen Benutzerbereichen läßt sich kopieren. Dabei ist nur bei beiden Dateinamen die betreffende G-Option ([Gn]) angegeben.

```
A>PIP PROG5[G5]=PROG12[G12]
```

Während »G« bei der Zieldatei die einzige erlaubte Option ist, gibt es für die Quelldateien eine Reihe zusätzlicher Anweisungen, die noch zur Sprache kommen.

Mehrere Dateien vermag PIP.COM zu einer einzigen zusammenzufassen.

Befehle en masse

Dazu listet man die einzelnen Namen der Quelldateien durch Kommas getrennt auf. Haben Sie zum Beispiel ein Buch geschrieben, das aus den Artikeln 1 bis 5 besteht, können Sie diese in der Datei »BUCH« vereinigen.

```
A>PIP BUCH=TEXT1,TEXT2,TEXT3,
TEXT4,TEXT5
```

Soweit der Übertrag zwischen den diskettenorientierten Geräten.

Die zeichenorientierten Geräte heißen unter CP/M 2.2 »CON:«, »LST:«, »RDR:« und »PUN:«. »CON:« und »LST:« stehen auch unter CP/M Plus weiterhin zur Verfügung. »CON:« (für »Console«) bezeichnet sowohl den Bildschirm als auch die Tastatur. Die Unterscheidung trifft das Betriebssystem dadurch, daß Zeichen von der Tastatur nur gelesen und auf den Bildschirm nur geschrieben werden können.

Wollen Sie beispielsweise eine Datei von der Diskette lesen und auf dem Bildschirm auflisten (entsprechend dem TYPE-Befehl), gilt »CON:« als Zielgerät und wird somit als Bildschirm identifiziert.

```
PIP CON:=KEYS.CCP
```

In einem anderen Artikel dieser Ausgabe (»Spezialitäten«) sollten Sie PIP.COM als »Mini-Editor« benutzen. Dort stand »CON:« auf der rechten Seite des Gleichheitszeichens und bezeichnete damit die Tastatur:

```
PIP DATEI=CON:
```

Es werden so lange Zeichen von der Tastatur entgegengenommen und in die Datei »DATEI« geschrieben, bis Sie mit CTRL-Z unterbrechen. Um in eine neue Zeile zu gelangen, drücken Sie RETURN und CTRL-J.

Weniger kompliziert verläuft die Zuordnung von »LST:« zum Drucker. Denn ein Drucker kann immer nur ein Zielgerät sein. Die Ausgabe einer Datei auf dem Drucker erfolgt normalerweise mit CTRL-P und TYPE. Doch PIP.COM übernimmt auch dies.

```
PIP LST:=DATEI
```

Eine sehr einfache »Schreibmaschine« erhält man durch Zuordnung der Tastaturabfrage zur Druckerausgabe:

```
PIP LST:=CON:
```

Wie auf einer »echten« mechanischen Schreibmaschine ist aber die Korrektur des geschriebenes Textes unmöglich.

Auch die Bildschirmausgabe der Texteingabe ist erlaubt, doch wohl kaum sinnvoll.

```
PIP CON:=CON:
```

Das linke »CON:« steht für den Bildschirm, das rechte für die Tastatur.

Unter CP/M 2.2 gibt es zwei zusätzliche Datengeräte, die ebenso über PIP.COM angesprochen werden können. Sie stammen sozusagen aus der Computer-»Steinzeit«. »RDR:« bezeichnet den Lochstreifenleser (»Reader«) und »PUN« den dazugehörigen Stanzer (»Puncher«). Unter CP/M Plus entfallen diese Abkürzungen. Da die Routinen fast ausschließlich zur Steuerung serieller Schnittstellen benutzt werden, ging man von den speziellen Namen weg und wählte die allgemeingültige Bezeichnung »AUX:« für »Auxiliary Device« (auf Deutsch »Hilfseingang« beziehungsweise »Hilfsausgang«). Ähnlich wie bei »CON:« wird die Unterscheidung zwischen der Ein- und Aus-



gabe getroffen. Beim Schneider sind die AUX:-Geräte bisher nicht belegt – also noch frei für eigene Programme.

Neben diesen Geräten, die in der CP/M-Terminologie »logische Geräte« heißen, können noch einige spezielle Geräte nur von PIP.COM angesprochen werden.

So darf anstelle von »LST:« auch »PRN:« stehen. Die beiden Bezeichnungen gleichen sich darin, daß sie die Druckerausgabe steuern. »PRN:« numeriert aber automatisch die Zeilen, setzt Haltepunkte für den Tabulator in jeder achten Spalte und erzeugt alle 60 Zeilen einen Seitenvorschub (Form-Feed).

Der Mehrfach-kommando-Modus

Speziell für Lochstreifenleser gedacht ist »NUL:«. 40 vorangehende hexadezimale Nullen werden von PIP.COM überlesen.

Eine weitere Eingabeeinheit heißt »EOF:«. Sie entspricht »AUX:«, sendet aber als Textende-Markierung den ASCII-Code 26 (CTRL-Z).

Bleiben noch zwei Geräte übrig. Von diesen läßt Digital Research in der CP/M-Plus-Dokumentation absolut nichts verlauten. Es handelt sich um die benutzerdefinierbaren Einheiten »INP:« und »OUT:«. Sie sind aber auch nicht zur Änderung durch den Anwender vorgesehen – höchstens durch versierte Programmierer. Denn dazu muß PIP.COM gepatcht werden. Damit läßt sich beispielsweise eine RAM-Disk in das CP/M-Betriebssystem integrieren.

Das etwas seltsame Wort »Mehrfachkommando-Modus« stammt aus der Original-Dokumentation von Digital Research und ist ein typisches Beispiel für eine manchmal etwas holprige Übersetzung. Im englischen Originaltext lautet der Ausdruck »Multiple Command Mode«. Er besagt ganz einfach, daß Sie PIP.COM nicht jedesmal neu laden müssen, wenn Sie eine ganze Reihe von Dateien zu kopieren haben. Völlig überflüssig (und vor allem langwierig) ist beispielsweise die Befehlsfolge:

```
A>PIP CON:=FILE
A>PIP LST:=DATEI
A>PIP DATEI=B:FILE
A>PIP LST:=CON:
```

Der Computer muß hierbei den Programmcode von PIP.COM insgesamt viermal in den Speicher laden. Dabei erübrigt sich dies völlig, denn PIP befindet sich nach jedem Aufruf weiterhin im RAM.

Bedeutend einfacher macht es der »Multiple Command Mode«. Er wird aufgerufen, indem Sie PIP ohne einen einzigen Parameter eingeben. Sobald PIP geladen ist, erscheint die Bereitschaftsmeldung, und Sie können die Befehle eingeben. Das Nachladen von PIP.COM entfällt. Abgebrochen wird PIP mit CTRL-C oder durch Drücken der RETURN-Taste.

```
A>PIP
CP/M 3 PIP VERSION 3.0
*CON:=FILE
*LST:=DATEI
*DATEI=B:FILE
*LST:=CON:
*^C
```

Wie Sie sicher wissen, arbeiten die Programme unter CP/M 3.0 oft mit verschiedensten Optionen. Denken Sie zum Beispiel an DIR.COM. Auch PIP.COM kennt eine Reihe von einbuchstabigen Befehlen, die das Übertragen der Dateien beeinflussen. Manche Optionen von PIP unter CP/M 2.2 entfielen. Sie kommen auch in der Dokumentation von Digital Research nicht mehr zur Sprache.

Weiterhin vorhanden sind **[Dn]** (löscht alle Zeichen ab der Spalte n), **[E]** (Echo aller übertragenen Texte auf der Konsole) und **[F]**, das die Ausgabe von Form-Feed unterdrückt. Ebenso wird **[H]** (Überprüfung von Intel-Hex-Dateien auf korrekten Aufbau) und [I] (alle :00-Datensätze in Intel-Hex-Files unterdrückt) unterstützt. **[L]** wandelt die Großbuchstaben einer Datei in kleine um. **[U]** macht das Gegenteil. Mit **[N]** schreibt PIP.COM vor jede Zeile eine Zeilennummer. Eine Abwandlung davon ist der **[N2]**-Befehl, der vor die Zeilennummer führende Nullen setzt.

Sollen Objektcode-Dateien von Maschinenprogrammen mit PIP.COM verkettet werden, darf der Computer das Dateiende nicht mit Hilfe von CTRL-Z abfragen. Denn dieses Zeichen kommt in ganz normalen Maschinencode-Programmen vor – LD A,(DE) im Z80-Code beziehungsweise LDAX D als 8080-Mnemonic. Damit die Verknüpfung dennoch klappt, muß der Befehl **[O]** (»Object file transfer for machine code«) hinzugefügt werden.

Die Page-Option **[Pn]** führt alle n Zeilen einen Seitenvorschub aus. **[Tn]** weist den Computer an, alle Tabulator-Zeichen (CTRL-I) durch die entsprechende Anzahl von Leerzeichen zu ersetzen. **[R]** erlaubt es, SYS-Dateien zu lesen.

Um nur bestimmte Teile einer Datei zu extrahieren, finden **[Q]** und **[S]** Verwendung. »Q« steht für »Quit copying after string found« und »S« für »Start copying from string«. Stellen Sie sich vor, in der Datei ARNOLD.TXT stünde der Text von Bild 1.

Textkomfort durch viele Befehle

Wollen Sie die Adresse von Arnold Schneider in eine Kundendatei einer Firma aufnehmen, können Sie diese mit PIP.COM aus ARNOLD.TXT »herausschneiden«.

CP/M wandelt Texte in der Kommandozeile grundsätzlich in Großbuchstaben um. Als letztes zu übernehmendes Wort suchen wir aber nicht »AMSTRADAM«, sondern »Amstradam«. Deshalb greifen wir auf einen kleinen Trick zurück und verwenden den Direkt-

```
Herrn
Arnold Schneider
Triumphstraße 464

6128 Amstradam


Sehr geehrter Herr Schneider,

anbei erhalten Sie die neueste Version unseres
Textverarbeitungsprogramms. Wir wünschen Ihnen
viel Spaß damit.


Mit freundlichen Grüßen


A>
```

Bild 1. Die Datei »ARNOLD.TXT«

```
Sehr geehrter Herr Schneider,

anbei erhalten Sie die neueste Version unseres
Textverarbeitungsprogramms. Wir wünschen Ihnen
viel Spaß damit.

Mit freundlichen Grüßen
```

Bild 2. Die Ausgabe der Datei ARNOLD.TXT nach dem Wort »Sehr«

A	-	Archive files copy Es werden nur Dateien kopiert, deren Datumsattribut zeigt, daß sie seit dem letzten Backup verändert wurden.
C	-	Confirm files copy PIP.COM fragt den Benutzer vorher, ob er eine spezielle Datei überhaupt kopieren möchte.
Dn	-	Delete from column n Beim Übertrag werden alle Zeichen rechts der Spalte n gelöscht.
E	-	Echo transfer at console Alle Zeichen werden beim Übertrag zusätzlich auf dem Bildschirm ausgegeben.
F	-	Filter form feeds Entfernt beim Übertrag alle Form-Feed-Zeichen aus der Datei.
Gn	-	Get from user n/Go to user n Holt die Datei aus dem Benutzerbereich n oder überträgt sie in diesen.
H	-	Check Intel-Hex files Intel-Hex-Dateien werden auf Fehlerfreiheit überprüft. Wenn sie nicht den Vorschriften entsprechen, erscheint eine Meldung.
I	-	Ignore :00 records Ignoriert alle Aufzeichnungen einer Intel-Hex-Datei, die mit »:00« beginnen.
L	-	Translate to lower case Alle Großbuchstaben einer Datei werden in der Zieldatei in Kleinbuchstaben umgewandelt.
N	-	Add line numbers Fügt vor jeder Zeile in der Ausgabedatei eine Zeilennummer ein.
N2	-	Add line numbers (second version) Wie [N], jedoch werden die Zeilennummern mit führenden Nullen geschrieben.
O	-	Object file transfer for machine code Kopiert Dateien, ohne das Dateiende-Kennzeichen CTRL-Z zu beachten. Dient zur Verkettung von Maschinencode-Programmen.
Pn	-	Insert Page breaks after every n lines Alle n Zeilen wird in die Ausgabedatei ein Form-Feed-Zeichen (ASCII-Code 12) eingefügt.
Qtext^Z	-	Quit copy after string »text« Kopiert die Datei, bis der String »text« entdeckt wird. Dieser wird noch mitübertragen.
R	-	Read files with SYS-attribute Kopiert auch Systemdateien. Das sind Dateien, die mit dem SYS-Attribut versehen sind.
Stext^Z	-	Start copy from string »text« Beginnt das Kopieren der Datei an der Stelle, an der der String »text« gefunden wird.
Tn	-	Expand tabulators Ersetzt alle TAB-Zeichen (CTRL-I) durch Leerzeichen. Deren Zahl wird durch »n« bestimmt.
U	-	Convert to upper case Alle Kleinbuchstaben in der Datei werden in Großbuchstaben umgewandelt.
V	-	Verify that data has been copied correctly Überprüft die an das Peripheriegerät übertragenen Zeichen nochmals. Das ist nur bei Diskettendateien möglich.
W	-	Write over files with R/O attribute Überschreibt auch schreibgeschützte Dateien ohne Warnung und Sicherheitsabfrage.
Z	-	Set highbit to zero Setzt in allen Zeichen der Zieldatei das siebte Bit auf den Wert Null.

Bild 3. Die Fähigkeiten von PIP im CP/M 3.0

eingabe-Modus, der Klein- und Großbuchstaben akzeptiert.

```
A>PIP
*CON:=ARNOLD.TXT[QAmstradam^Z]
 Herrn
 Arnold Schneider
 Triumphstraße 464

 6128 Amstradam
 *^C
```

Der String muß also mit CTRL-Z abgeschlossen werden. Das gilt auch bei **[S]**. Dieses kopiert den Text beginnend mit der Stelle, an der der String gefunden wird. Um beispielsweise allein den Brief, diesmal ohne Adresse, zu zeigen, kann man nach dem Wörtchen »Sehr« suchen.

```
A>PIP
 *CON:=ARNOLD.TXT[SSehr^Z]
```

Die Ausgabe finden Sie in Bild 2. PIP verlassen wir mit CTRL-C.

Durch eine Kombination der S- und der Q-Option können dann aus einer Datei einzelne Datensätze herausgezogen werden.

[V] (Verify) gestattet die Überprüfung von Daten nach dem Übertrag. Dies ist aber nur möglich, wenn das Ziel eine Diskettendatei ist.

[W] Überschreibt schreibgeschützte Dateien (R/O-Files) ohne Anfrage an den Benutzer und **[Z]** setzt das siebte Bit aller Zeichen auf Null. Diese Option wird vor allem für Wordstar-Dateien benutzt, bei denen das höchstwertige Bit eines Zeichens als Kennung für das Ende eines Wortes steht.

Kombinationsmöglichkeiten

Selbstverständlich lassen sich die Optionen – soweit sinnvoll – miteinander kombinieren. Um beispielsweise in einer Wordstar-Datei das siebte Bit zu löschen, alle Kleinbuchstaben in große zu wandeln und die Aufzeichnung zu überprüfen, muß man

```
A>PIP DATEI=WSTAR.DAT[ZUV]
```

eingeben.

Neu unter CP/M-3.0 ist **[C]** – die Confirm-Option – die bei jedem Dateinamen, der auf die Eingabe paßt, erst einmal eine Sicherheitsfrage an den Benutzer richtet, ob die Datei überhaupt kopiert werden soll. Ebenso gibt es jetzt das Archiv-Attribut **[A]**, das es erlaubt, nur diejenigen Dateien zu kopieren, die seit dem letzten Backup verändert wurden – das Festplatten-Backup läßt grüßen. Es funktioniert aber nur, wenn das Archiv-Bit im Dateinamen von SET.COM (siehe »STAT« ade!« in diesem Heft) beeinflußt wurde. Was PIP alles kann, zeigt Bild 3.

(Martin Kotulla/hg)



STAT ade!

Das beliebte Allzweck-Programm STAT.COM von CP/M 2.2 gehört nicht zum Lieferpaket CP/M Plus. Es wurde durch die erheblich leistungsfähigeren Programme DEVICE.COM, SET. COM und SHOW.COM ersetzt.

Welcher CP/M-Freak kennt es nicht – STAT.COM, ein Universalprogramm für fast jeden Zweck. Nun ging es den Weg aller überholten Software und wurde ausrangiert. Es hätte aber auch einige Schwierigkeiten gehabt, mit den neuen Routinen von CP/M Plus zurechtzukommen. Bevor Sie jetzt dem Programm nachtrauern: Die Nachfolgeprogramme DEVICE.COM, SET. COM und SHOW. COM sind erheblich leistungsfähiger geworden und ersetzen STAT.COM angemessen.

Beginnen wir mit SET.COM. Es besitzt eine Reihe von Befehlen, die einzelne Dateien oder ganze Disketten mit Attributen versehen. Es können Paßwörter vereinbart und Disketten benannt werden. So legt »SET d:filename.typ [Attributsliste]« eines oder mehrere der folgenden Attribute für eine Einzeldatei oder eine Dateigruppe fest.

– SET DATEI [SYS] versieht das Feld »DATEI« mit dem SYS-Attribut. Solche Felder werden nicht mehr mit DIR, sondern nur mit DIRSYS gelistet. Ist eine SYS-Datei im Userbereich 0 gespeichert, kann sie von allen anderen Benutzerbereichen direkt aufgerufen werden:

```
A>SET PIP.COM [SYS]
A:PIP.COM set to system (SYS),
Read Write (RW)
A>USER 9
9A>PIP
CP/M 3 PIP VERSION 3.0
*
```

– Im Gegenzug schaltet SET DATEI [DIR] das SYS-Kennzeichen ab. Die Datei erscheint wieder im Directory, ist aber auch nicht mehr von fremden Userbereichen aufzurufen.

– SET DATEI [RO] versieht die angegebene Datei mit einem Schreibschutz. Die Datei ändert sich zu »Read-Only« und kann nur noch gelesen werden.

```
A>SET PIP [RO]
A:PIP.COM set to system (SYS),
Read Only (RO)
```

– SET DATEI [RW] macht die letzte Anweisung wieder rückgängig. Um beispielsweise bei PIP.COM beide Attribute (RO und SYS) abzuschalten, müssen die beiden Bezeichnungen kombiniert werden.

```
A>SET PIP.COM [RW,DIR]
A:PIP.COM set to directory (DIR),
Read Write (RW)
```

– Die Archiv-Option von PIP.COM ist auch über SET zu steuern. Dateien mit dem Merkmal [ARCHIVE=OFF] kopiert PIP.COM bei der A-Option, die mit [ARCHIVE=ON] nicht. PIP setzt nach einem erfolgreichen Kopiervorgang das Archiv-Bit auf ARCHIVE=ON.

– Es gibt vier benutzerdefinierbare Dateiattribute. Sie tragen die Namen F1, F2, F3 und F4. Der Benutzer kann sie nach eigenem Gutdünken verwenden. Sinnvoll ist das aber nur, wenn ein Programm speziell für CP/M 3.0 geschrieben ist und darauf zurückgreift. Die Attribute werden gesetzt mit [Fn=ON] und gelöscht mit [Fn=OFF].

```
A>SET PIP.COM [F1=OFF,F2=ON,
F3=OFF,F4=ON]
A:PIP.COM set to directory (DIR),
Read Write (RW) 24
```

Sie erkennen an der Zahl »24«, daß das F2- und das F4-Attribut angeschaltet sind, F1 und F3 hingegen nicht.

Auch Laufwerksattribute dürfen vereinbart werden. So lassen sich Diskettenlaufwerke mit [RO] auf »Read-Only« setzen. Damit sind alle schreibenden Diskettenzugriffe zum Scheitern verurteilt. Abgeschaltet wird der Schutz mit [RW]. Er ist übrigens stets bis zum nächsten Warmstart, zum Beispiel mit CTRL-C, wirksam.

```
A>SET [RO]
Drive A: set to Read Only (RO)
A>SET B:[RW]
Drive B: set to Read Write (RW)
```

Eine weitere Routine von SET.COM ist relativ unbekannt. SET erlaubt es, der Diskette einen Namen (ein »Label«) zuzuordnen (siehe Bild 1).

Mit SHOW [LABEL] kann der Name später wieder abgefragt werden.

Falls Sie besonders wertvolle und schutzwürdige Informationen besitzen, etwa die Liste aller Ihrer Freundinnen, können Sie ein Paßwort vereinbaren, das wahlweise für einzelne Dateien oder die ganze Diskette gilt. Zuerst der Schutz der kompletten Diskette. Mit

```
A>SET [PASSWORD=SECRET]
```

teilen Sie dem SET.COM-Programm mit, daß es auf der Diskette einen Vermerk mit diesem Paßwort anbringen soll. Versucht ein anderer Benutzer, Attribute der Diskette mit SET.COM zu verändern, ist er gezwungen, das Paßwort einzugeben. Beispielsweise ist das Umbenennen der Diskette ohne Paßwort unmöglich, wie folgender Dialog mit dem Computer beweist:

```
A>SET [PASSWORD=SECRET]
Password = SECRET
A>SET [NAME=NEUER NAME.DSK]
Directory Label
Password? DONTKNOW
ERROR: Wrong Password
A>
```

Die Eingabe des Paßworts bei der Sicherheitsabfrage geschieht im »Blindflug«. Sie sehen also nicht, was Sie eingeben. So kann Ihnen weder jemand über die Schultern schauen, noch mit dem Protokoll (mit Hilfe von CTRL-P) etwas anfangen.

Ist eine Datei auch vor dem Laden geschützt, hängt man das Paßwort an den Dateinamen – durch einen Strichpunkt von diesem getrennt.

```
A>PIP;SECRET
```

Das Geheimwort kann man natürlich auch wieder aufheben. Dazu geben Sie nach »PASSWORD=« einfach RETURN oder ENTER ein. Der Computer fragt dann das Paßwort ab. Ist es korrekt, wird der Schutz aufgehoben.

```
A>SET [PASSWORD=
Directory Label
Password? SECRET
```

Auch einzelne Dateien lassen sich schützen, und zwar wahlweise vor dem Lesen, Schreiben oder Löschen. Bevor

```
A>SET [NAME=CPM3SYS.DSK]

Label for drive A:

Directory      Passwds Stamp   Stamp   Stamp
Label          Reqd    Create  Access  Update
-------------- ------- ------- ------- ------
A:CPM3SYS .DSK   off     off     off     off
```

Bild 1. Taufen Sie Ihre Diskette

```
A>SET [PROTECT=ON]

Label for drive A:

Directory      Passwds Stamp   Stamp   Stamp
Label          Reqd    Create  Access  Update
-------------- ------- ------- ------- ------
A:CPM3SYS .DSK   on      off     off     off
```

Bild 2. Programmschutz gewünscht?

Sie das angehen, müssen Sie zuerst dem Betriebssystem mitteilen, daß ein Programmschutz gewünscht ist. Diese Information übernimmt [PROTECT=ON] (Bild 2).

Sie sehen die Veränderung in der Spalte »Passwords required«. Dort steht jetzt »on«.

[PROTECT=OFF] schaltet diese Schutzvorbereitung wieder ab. War bereits vorher ein Schutzwort für die gesamte Diskette vereinbart, so muß dieses bei beiden PROTECT-Anweisungen dem System mitgeteilt werden. Der Computer fragt gegebenenfalls danach.

»PROTECT=ON« aktiviert den Schutz für alle Dateien. Dazu geben Sie den (ein- oder mehrdeutigen) Dateinamen und das Paßwort ein (Bild 3).

```
A>>SET *.COM [PASSWORD=COM]
A:SUBMIT  .COM  Protection=READ, Password=COM
A:SETKEYS .COM  Protection=READ, Password=COM
A:LANGUAGE.COM  Protection=READ, Password=COM
A:SET24X80.COM  Protection=READ, Password=COM
```

Bild 3. Alle Dateien hören auf »COM«

Dieser Befehl schützt alle Kommandodateien der Diskette mit dem Paßwort »COM«. Sehen Sie sich aber bei der Anwendung vor, denn wenn Sie SET.COM ebenfalls schützen, stehen Sie auch als rechtmäßiger Benutzer vor verschlossenen Türen und können den Schutz der Programme nicht mehr aufheben. Also alle Versuche nur an einer Kopie vornehmen!

Es lassen sich verschiedene Arten des Dateischutzes angeben: READ verbietet das Lesen, Beschreiben, Löschen und Umbennenen einer Datei. WRITE entspricht READ, Lesen ist jetzt erlaubt. DELETE verhindert lediglich das unbefugte Löschen und Umbenennen von Dateien. NONE hebt den Schutz wieder vollständig auf. Soll beispielsweise bei PIP.COM nur das Löschen verhindert werden, müssen Sie folgendermaßen vorgehen:

```
A>SET [PROTECT=ON]
A>SET PIP.COM [PASSWORD=PIPCOM]
A:PIP.COM Protection = READ,
Password = PIPCOM
A>SET PIP.COM [PROTECT=DELETE]
A:PIP.COM
Password? PIPCOM
A:PIP.COM Protection = DELETE
```

Danach kann PIP.COM ganz normal geladen und gestartet werden.

```
A>PIP
CP/M 3 PIP VERSION 3.0
*^C
```

Nur das Löschen, beispielsweise mit ERA, ist jetzt unmöglich.

```
A>ERA PIP.COM
A:PIP.COM Not erased,
Password Error
Password: ........
A:PIP.COM Not erased,
Password Error
A>
```

Wie fragwürdig diese ganze »Schützerei« jedoch ist, werden Sie bemerken, wenn Sie Basic aufrufen, um von dort aus mit Hilfe von IERA die Datei zu löschen.

```
a$="PIP.COM":IERA, @A$
Ready
```

Unter Basic ist der Schutz nicht wirksam, da Amsdos Byte für Byte bei allen drei Schneider-Computern übereinstimmt und mit CP/M 2.2 abgestimmt ist. Und dieses Betriebssystem kennt noch keine Paßwörter. Wenn Sie aber nur unter CP/M Plus arbeiten, dann haben die Paßwörter Sinn.

Eine Spezifikation kam noch nicht zur Sprache und zwar [DEFAULT=password]. Damit lassen sich all die Dateien schützen, denen nicht ausdrücklich ein eigenes Kennwort zugewiesen wurde.

```
A>SET [DEFAULT=GEHEIM]
Default password = GEHEIM
```

SET.COM kennt auch die Datums- und Zeitmarkierung von Dateien. Wie, das lesen Sie in dem Artikel »Sekundengenau« in dieser Ausgabe.

Was Sie von STAT.COM unter CP/M 2.2 noch gewohnt sind, was aber bei SET.COM fehlt, das finden Sie fast alles in SHOW.COM wieder. SHOW.COM dient als »Anzeigeprogramm«. Es offenbart diverse Daten über die Disketten beziehungsweise die Laufwerke. So gibt SHOW [SPACE] den noch freien Speicherplatz des angemeldeten Laufwerks und den Diskettenstatus an.

```
A>SHOW [SPACE]
A: RW, Space: 18k
```

Zusätzlich darf vor der eckigen Klammer ein Laufwerksname spezifiziert werden.

```
A>SHOW A:[SPACE]
A>SHOW B:[SPACE]
```

Die Kennung [SPACE] kann aber auch wegfallen, da es die Standardform von SHOW.COM ist.

```
A>SHOW
A: RW, Space: 18k
```

Mit Anweisung zeigt SHOW den freien Speicherplatz aller Laufwerke, sofern seit dem Systemstart mindestens ein Zugriff darauf folgte.

```
A>B:
B>A:
A>SHOW
A: RW, Space: 18k
B: RW, Space:  1k
```

SHOW [LABEL] eröffnet Ihnen auch, welchen Namen die Diskette besitzt. Die Zuweisung des Namens übernimmt SET.

```
A>SET [NAME=CPM3SYS.DSK]
A>SHOW [LABEL]
Directory Label: CPM3SYS.DSK
```

Bei dieser Gelegenheit gibt SHOW.COM noch weitere Informationen über die Diskette aus:

– »Passwds Reqd« (»Passwords required«) gibt an, ob die Dateien mit Kennwörtern geschützt sind.

– »Stamp Create« und »Stamp Update« sind für die Datumsangabe vorgesehen und zeigen an, ob diese aktiv ist.

– »Label Created« gibt das Datum und die Uhrzeit des Zeitpunkts aus, an dem die Diskette den Namen erhielt. Da die Uhr beim Schneider nicht akku- oder batteriegepuffert ist, setzt der Computer jedesmal den 15. Dezember 1982, 00:00 Uhr Mitternacht ein, sofern die Uhr nicht mit DATE.COM eingestellt wird.

– Ähnliches gilt für »Label Updated«, das den Zeitpunkt der letzten Änderung des Namens angibt.

SHOW [DRIVE] listet wie bei C STAT DSK: unter CP/M 2.2 das Format der Diskette auf. Hinzugekommen ist noch die Angabe »Bytes/Physical Record«. Bild 4 zeigt einen Ausdruck.

SHOW [DIR] gibt an, wieviele Einträge im Inhaltsverzeichnis der Diskette noch frei sind.

```
A>SHOW [DIR]
A: Number of free directory
entries: 30
```

Mit SHOW [USERS] erfahren Sie alles, was mit den verschiedenen Benutzerbereichen zusammenhängt, in welchen Bereichen Dateien stehen, wieviele das im einzelnen sind und zusätzlich die Zahl der freien Directory-Einträge im Gesamtverzeichnis.

```
A>SHOW [USERS]
A: Active User :   0
A: Active Files:   0   6
A: # of files  :  28   1
A: Number of free directory
entries: 33
```

Aus dieser Ausgabe können Sie folgende Schlüsse ziehen: Der augenblicklich aktive Benutzerbereich ist 0 (»Active User«). Dateien stehen im Bereich 0 und User 6 (»Active Files«). Unter User 0 befinden sich 28 Verzeichniseinträge, im Bereich 6 nur einer (»# of files« heißt »number of files«). Das Inhaltsverzeichnis kann noch 33 Einträge aufnehmen (»Number of free directory entries«).

Doch STAT.COM leistet noch mehr. Es weist auch den logischen Peripheriegeräten physikalische zu und zeigt diese an. So läßt sich etwa der logische

```
A>SHOW [DRIVE]

     A: Drive Characteristics
 1,368: 128 Byte Record Capacity
   171: Kilobyte Drive  Capacity
    64: 32 Byte  Directory Entries
    64: Checked  Directory Entries
   128: Records / Directory Entry
     8: Records / Block
    36: Sectors / Track
     2: Reserved  Tracks
   512: Bytes / Physical Record
```

Bild 4. So zeigt sich die Diskette von ihrer besten Seite

```
Physical Devices:
I=Input,O=Output,S=Serial,X=Xon-Xoff
CRT      NONE  IO       LPT      NONE  O

Current Assignments:
CONIN:    = CRT
CONOUT:   = CRT
AUXIN:    = Null Device
AUXOUT:   = Null Device
LST:      = LPT

Enter new assignment or hit RETURN
```

Bild 5. Logische und physikalische Geräte unter CP/M Plus

LST:-Kanal auf ein serielles Interface umlenken.

Das erledigt unter CP/M Plus DEVICE.COM. »DEVICE« (ohne Parameter aufgerufen) listet die aktuellen Zuweisungen auf. Und da hat sich einiges geändert – sowohl auf Benutzer- als auch auf Programmiererebene. Eine Ausgabe von DEVICE finden Sie im Bild 5.

DEVICE.COM versteht mehrere Namen für jedes logische Gerät, was die Eingabe neuer Namen sehr erleichtert. Die »CON:«-Konsole teilt sich in »CONIN:« (für die Tastaturabfrage) und »CONOUT:« (für die Bildschirmausgabe). Dennoch kann »CON:« auch weiterhin allein benutzt werden – ebenso »CONSOLE:«. »CONIN:« läßt sich auch mit »KEYBOARD:« aufrufen. »AUX:« wird auf Wunsch zu »AUXILIARY:«, »AUXIN:« und »AUXOUT:«. Das »LST:«-Gerät wird mit »PRINTER:« angesprochen.

Die Liste der physikalischen Geräte ist nicht starr festgelegt, sondern vom jeweiligen System abhängig. Der nicht ausgebaute CPC 6128 besitzt deshalb nur »LPT« (»Line Printer«, über eine parallele Schnittstelle angeschlossener Drucker) und »CRT« (»Cathode Ray Tube«, Bildschirm). Beachten Sie bitte, daß CP/M Plus (anders als CP/M 2.2) die Geräte ohne Doppelpunkt beschreibt.

Um dem logischen Gerät »LST:« das physikalische »CRT« zuzuweisen, geben Sie

```
A>DEVICE LST:=CRT
```

ein. Damit werden alle Ausgaben zum Drucker auf den Bildschirm geschrieben. Der Drucker leidet dann allerdings an mangelnder Arbeitsauslastung. Schalten Sie ihn jedoch mit CTRL-P an und geben Sie »DIR« ein. Sie werden eine Überraschung erleben. Gott sei Dank verschwindet dieser Spuk durch ein weiteres CTRL-P schnell wieder.

Ein besonderes Gerät ist das »Null-Device«. Wenn zum Beispiel ein CP/M-Programm Texte an den Drucker schickt und es ist keiner angeschlossen, so läßt sich die Druckerausgabe damit abschalten. Der Computer wartet nicht mehr, bis er das einzelne Zeichen abgesetzt hat.

```
A>DEVICE LST:=NULL
```

Der Aufruf CTRL-P ist nun völlig wirkungslos.

Diese Technik benutzt auch das Schneider-BIOS, nachdem es festgestellt hat, daß überhaupt kein Drucker vorhanden ist. Geben Sie nämlich »DEVICE« ein, so steht dort unter anderem:

```
LST: = LPT
```

Drücken Sie jetzt CTRL-P, geschieht gar nichts mehr, weil der Computer versucht, ein Zeichen an den Drucker abzuschicken. Gelingt ihm das nicht, erscheint nach einigen Sekunden in der untersten Bildschirmzeile eine Fehlermeldung:

```
LPT not ready - Retry,
Ignore, Cancel
```

Nach »C« für »Cancel« ist jeder CTRL-P-Befehl unwirksam. Schauen Sie sich mit DEVICE.COM wieder die Zuweisungstabelle an, erfahren Sie auch wieso.

```
LST: = Null Device
```

Das Betriebssystem hat damit den Drucker automatisch als nicht-existent eingestuft.

Dieses Wissen läßt sich sinnvoll ausnutzen. Haben Sie beispielsweise versehentlich bei der Fehlermeldung »C« für »Cancel« gedrückt, können Sie mit DEVICE den Drucker wieder empfangsbereit machen und die alte Zuweisung wiederherstellen.

```
A>DEVICE LST:=LPT
```

DEVICE.COM leistet aber noch mehr. So erfahren Sie beispielsweise mit »DEVICE NAMES« die Namen der physikalischen Geräte.

»DEVICE VALUES« zeigt die Zuweisung der logischen und physikalischen Geräte.

Auch die Charakteristika einzelner logischer und physikalischer Geräte können Sie ausgeben lassen.

```
A>DEVICE CON:
```

Die physikalischen Peripherie-Geräte, hauptsächlich serielle und parallele Schnittstellen, können mit DEVICE angepaßt werden. So schaltet DEVICE LPT [XON] das XON/XOFF-Protokoll ein und »NOXON« wieder ab. Die Baudrate läßt sich mit DEVICE.COM verändern. Gerade hierzu sollte man aber das Schneider-spezifische Programm SETSIO.COM vorziehen, weil solche Einstellungen primär von der Hardware abhängen.

Das Bildschirmformat kann DEVICE.COM ebenfalls anzeigen und verändern. Wieviele Spalten und Zeilen der Computer adressiert, zeigt »PAGE«.

```
A>DEVICE CONSOLE [PAGE]
Console width set to 79 columns
Console page set to 24 lines
```

Mit einem anderen DEVICE-Befehl läßt sich unter CP/M Plus ein »Bildschirmfenster« setzen:

```
A>DEVICE CONSOLE [COLUMN=40,
LINES=12]
```

Allerdings funktioniert das nicht korrekt, weil ab und zu auch einmal die rechte Fenstergrenze überschrieben wird. Dieser Fall tritt ein, wenn Sie eine Eingabezeile mit beliebigen Buchstaben vollschreiben und der Computer diese Eingabe als Fehlermeldung mit einem Fragezeichen wiederholt.

Auch fällt die »Lines«-Grenze anders aus, als der Befehl WINDOW im Schneider-Basic. Denn der außerhalb des Windows liegende Text wird mitgescrollt. Nur die Meldung »Press RETURN to Continue« erscheint entsprechend häufiger.

Durchaus sinnvoll ist aber der folgende DEVICE-Aufruf:

```
A>DEVICE CONSOLE [COLUMN=80]
```

Aus unerklärlichen Gründen werden normalerweise im 6128-CP/M nur 79 statt 80 Zeichen jeder Zeile ausgenutzt. Damit passen aber bei DIR nur jeweils vier Dateinamen in jede Bildschirmzeile. Nach diesem DEVICE-Aufruf sind es immerhin fünf.

(Martin Kotulla/hg)



Gegen die Routine

Gewisse Routineaufgaben fallen bei der Programmierung immer wieder an. CP/M Plus kann solche Aufgaben für Sie erledigen.

Wollen Sie beispielsweise von bestimmten Dateien im Laufwerk A eine Sicherheitskopie auf B anlegen, so müssen Sie folgende Befehle eingeben:

```
A>PIP B:=ADRESS.DAT
A>PIP B:=FAKTURA.COM
A>PIP B:=*.WS
A>DIR A:
A>DIR B:
```

Sollen Ihre Daten nach jeder Bearbeitung aktuell gesichert werden, dann müssen Sie jedesmal vor dem Ausschalten des Computers die gleiche Befehlsfolge eingeben. Und jeder Computerfan sinnt bei solchen Routineaufgaben natürlich nach Arbeitserleichterung. Man braucht eine Datei, die dem Computer die Tastatureingabe »vormacht«. SUBMIT.COM kann das.

Legen Sie sich mit einem Texteditor zur Übung eine Datei mit dem Namen KOPIE.SUB und dem gerade angegebenen Inhalt an. Die Namensendung ».SUB« ist unerläßlich, da sonst die ganze Sache nicht funktioniert. Weiter geben Sie

```
A>SUBMIT KOPIE
```

ein. Der Console Command Processor (CCP) holt sich nun seine Eingaben nicht von der Tastatur, sondern von der Diskette. Unser Problem ist gelöst.

Was macht man nun aber bei Dateinamen, die sich immer wieder ändern, beispielsweise wenn ein Programm (oder eine Anweisung) gemeinsam mit einer Datei (ein Beispiel: ERA Datei) aufgerufen wird? Dazu kennt SUBMIT.COM bis zu neun Stringvariable. Diese tragen die Namen $1 bis $9.

Sie geben dazu nach »SUBMIT KOPIE« mehrere Dateinamen an, beispielsweise »SUBMIT KOPIE WS.COM WSOVLY1.OVR WSMSGS.OVR PIP.COM«. Ihre zugehörige Submit-Datei sieht dann so aus:

```
PIP B:=$1
PIP B:=$2
PIP B:=$3
PIP B:=$4
DIR B:
```

Das Submit-Programm legt beim Aufruf auf der Diskette eine Zwischendatei mit der Extension ».$$$« an, in der die Variablen durch die gewünschten Namen ersetzt sind.

```
PIP B:=WS.COM
PIP B:=WSOVLY1.OVR
PIP B:=WSMSGS.OVR
PIP B:=PIP.COM
```

Zu diesen neun Stringvariablen kommt noch $0. $0 hat eine besondere Funktion. Diese Variable enthält den Namen der eigentlichen ».SUB«-Datei. Damit läßt sich in einer sehr kurzen Datei eine Endlosschleife aufbauen.

```
$0
```

ist der gesamte Inhalt der Datei »END LOS.SUB«. Durch »SUBMIT ENDLOS« nimmt das Unheil seinen Lauf, denn die Datei ENDLOS ruft sich immer wieder selbst auf. Abbrechen kann man die Schleife nur mit CTRL-C. Sie müssen aber unter Umständen die Taste längere Zeit drücken, bevor Ihr Auftrag Beachtung findet, da die Tastaturabfrage teilweise abgeschaltet ist.

Wenn weniger Namen angegeben sind, als Stringvariablen im Submit-File stehen, werden die überzähligen Variablen mit Leerstrings angefüllt. Geben Sie mehr Namen an als benötigt, so werden die überzähligen ignoriert.

Stehen in der ».SUB«-Datei echte Dollarzeichen, sind diese zur Unterscheidung von den Stringvariablen durch zwei Dollarsymbole zu ersetzen. Wer sich zum Beispiel partout nur temporäre (vorläufige) Dateien ansehen will, hat

```
DIR *.$$$$$$
```

für

```
DIR *.$$$
```

einzugeben.

Schwierigkeiten treten auf, wenn Sie Steuerzeichen in die Submit-Datei aufnehmen wollen. Mit einem Texteditor geht das nicht problemlos. Es gibt eine ganz einfache Lösung. Setzen Sie beispielsweise für CTRL-P das Potenzierungszeichen und ein »P« ein, so daß in der Datei die beiden Buchstaben »^P« stehen. Diese interpretiert dann SUBMIT.COM als CTRL-P. Echte Potenzierungssymbole erkennt man wieder an der Verdoppelung.

SUBMIT.COM kann nur Tastatureingaben in der CCP-Kommandoebene simulieren. In laufenden Programmen ist das leider nicht möglich. Unter CP/M 2.2 existiert das Programm XSUB.COM (»Extended Submit«), das in die erste Zeile der ».SUB«-Datei geschrieben wird. Es hat den Nachteil, sich ebenfalls auf der zu bearbeitenden Diskette zu befinden und belegt dort sowohl einen Directory-Eintrag als auch 1 KByte Speicherplatz. Unter CP/M Plus gibt es XSUB.COM nicht mehr. Simulierten Eingaben für ein laufendes Programm steht ein »<« voran. Um beispielsweise PIP.COM im »Multiple Command Mode« (Modus, bei dem PIP ständig im Speicher steht und nicht permanent nachgeladen werden muß) zu benutzen, beschreiben Sie die ».SUB«-Datei mit

```
PIP
<B:=A:WS.COM
<B:=WSOVLY1.OVR
<B:=WSMSGS.OVR
<
B:
DIR
```

Das Textverarbeitungsprogramm Wordstar widersetzt sich leider der Automatisierung mit SUBMIT.COM. Es beachtet die Eingaben nicht und stellt sich tot. Das liegt daran, daß das Programm neben der Tastaturabfrage auch den Tastaturstatus untersucht. Und diesen beeinflußt SUBMIT.COM nicht. Wenn Sie aber während der Bearbeitung der Submit-Datei beliebige Tasten drücken, dann erkennt Wordstar bei der Statusabfrage der Tastatur diese als aktiv, und die Zeichen der Submit-Datei erscheinen auf dem Bildschirm.

Auch den Aufruf von SUBMIT.COM können Sie sich erleichtern. Und zwar auf zwei verschiedene Arten. Bei der ersten geben Sie

```
A>SUBMIT
```

ohne weitere Parameter ein. Das Betriebssystem lädt in diesem Fall das Programm in den Speicher und fragt

```
CP/M 3 SUBMIT Version 3.0
Enter File to SUBMIT:
```

Jetzt geben Sie den Namen der ».SUB«-Datei und die Parameter ein.

Der zweite Weg ist noch einfacher. Sie dürfen nämlich auch den Programmnamen »SUBMIT« einfach weglassen, müssen dann aber unbedingt die Extension ».SUB« an den Namen der Datei anfügen.

```
A>DATEI.SUB
```

SETDEF.COM (ausführliche Beschreibung ebenfalls in diesem Heft) hebt aber auch diese Einschränkung auf:

```
A>SETDEF [ORDER=(COM,SUB)]
A>DATEI
```

Nur müssen Sie jetzt beachten, daß die Diskette kein Programm mit dem Namen »DATEI.COM« enthält. Das ist deshalb wichtig, da sonst dieses anstelle der Submit-Datei gestartet wird.

Was bei vielen anderen Betriebssystemen häufig als Hilfsmittel zum Kopierschutz gedacht ist, kann dem CP/M-Benutzer unnötige Arbeit abnehmen: der Autostart.

Schauen Sie sich einmal mit »DIR« das Inhaltsverzeichnis Ihrer Systemdiskette an. Dort finden Sie die Datei »PROFILE.ENG«. Nennen Sie diese mit »RENAME PROFILE.SUB=PROFILE.ENG« um. Wenn Sie nun nach dem Zurücksetzen des Computers CP/M Plus neu starten, wird die Datei automa-

tisch aufgerufen. Sie beinhaltet zwei Befehle, die die Tastatur und den Zeichensatz an das Betriebssystem anpassen.

```
SETKEYS KEYS.CCP
LANGUAGE 3
```

Sie können sich natürlich unter dem Namen »PROFILE.SUB« eine beliebige andere Datei anlegen. Denkbar wäre etwa der Autostart einer Textverarbeitung oder eines anderen Anwenderprogramms. Damit machen Sie Computer-Unkundigen das Einarbeiten leichter.

GET & PUT

Unter CP/M 2.2 erfahren die zeichenorientierten Peripheriegeräte wie LST: und CON: eine völlig andere Behandlung als die Diskettendateien. So können Sie zwar mit CTRL-P die Druckerausgabe einschalten und den Bildschirminhalt auf den Drucker lenken, aber es existiert keine Tastenkombination und kein Hilfsprogramm, das die Bildschirmausgabe in eine Diskettendatei umleitet.

Völlig gleichrangig, wie beispielsweise unter MS-DOS, sind die verschiedenen Geräte unter CP/M Plus zwar immer noch nicht, sie sind aber einander »ähnlicher« geworden.

Das CP/M-Programm GET.COM nutzt dies aus und simuliert die Tastatureingabe, indem es sich die Daten aus einer Diskettendatei holt. Das hat fast den gleichen Effekt, wie oben mit SUBMIT.COM. Aber die Lösung mit GET.COM ist weniger leistungsfähig, da keine Variablen benutzt werden können.

GET besitzt aber den unschätzbaren Vorteil, auch mit Programmen wie beispielsweise Wordstar zusammenzuarbeiten. So findet es beim Aufruf der automatisierten Textverarbeitung Verwendung. Ein Beispiel: Sie haben eine Datei, in der jede Zeile 65 Zeichen lang ist. Um die Länge auf 40 Zeichen umzuformatieren, müssen Sie sich normalerweise durch den ganzen Text mit CTRL-B hindurcharbeiten. Mit GET.COM kann das der Computer, ohne Ihre Hilfe, allein machen.

Die Anweisung an das Programm GET.COM erfolgt schon fast in englischer Umgangssprache:

```
A>GET CONSOLE INPUT FROM FILE d:
filename.typ
```

Wem das zuviel zu schreiben ist, der kann auch verkürzt

```
A>GET FILE d:filename.typ
```

eingeben. Alle Daten für das laufende Programm holt der Computer jetzt von der Diskettenstation.

Im CCP – also bei der Direkteingabe – arbeitet GET.COM nicht. Das muß durch den Befehl »SYSTEM« gesondert verlangt werden. [SYSTEM] »sagt« dem CCP also, er soll seine Eingaben aus der Datei holen.

Ein kleines Beispiel für einen Wordstar-Brief, der vom Computer geschrieben wird, finden Sie im folgenden. Beachten Sie, daß die CTRL-Codes über CTRL-P mit dem betreffenden Buchstaben im Wordstar-Editor angegeben werden. Steht im Listing beispielsweise »^K«, müssen Sie CTRL-P und »K« drücken. Tippen Sie zu Testzwecken die folgende Datei im N-Modus (»Bearbeiten einer Programmdatei«) ein.

```
WS
DBRIEF
Sehr geehrte Damen und Herren,
bitte senden Sie uns 30 Computer
des Typs CPC 6128.
Mit freundlichen Grüßen
^KD
P^R
<RETURN>
<RETURN>
<RETURN>
<RETURN>
<RETURN>
<RETURN>
<RETURN>
X
```

Geben Sie dann

```
A>GET CONSOLE INPUT FROM FILE
AUTO.WS[SYSTEM]
```

oder

```
A>GET FILE AUTO.WS[SYSTEM]
```

ein.

Wordstar »regt sich zwar ganz fürchterlich über die Geschwindigkeit auf«, mit der die Zeichen von der Diskette geliefert werden und druckt haufenweise Ausrufezeichen aus, die den Anwender auffordern sollen, langsamer zu tippen, aber es gehen – zumindest bei diesem Text – keine Zeichen verloren. Dies beweist der selbsttätig erfolgte Ausdruck.

Neben »SYSTEM« gibt es noch weitere Steuerbefehle für die Bildschirmausgabe. »ECHO« fordert GET auf, alle Zeichen von der Diskette nicht nur an das Programm zu übergeben, sondern zusätzlich auch auf den Bildschirm zu schreiben. »NO ECHO« unterbindet dieses. »ECHO« ist der vorgegebene Standardwert.

Es gibt zwei Fälle, in denen die Übertragung von der Diskette abgebrochen wird. Entweder wurde die Datei bis zum Ende gelesen oder in der Datei steht ein Rückstellbefehl: »GET CONSOLE INPUT FROM CONSOLE« oder »GET CONSOLE«.

Das Gegenstück zu GET heißt »PUT.COM«. Mit diesem Programm leitet man die Ausgabe auf folgende Peripheriegeräte um.

```
A>PUT CONSOLE OUTPUT TO FILE d:
filename.typ
A>PUT CONSOLE FILE d:
filename.typ
```

Das veranlaßt den Computer, die Bildschirmausgabe in die angegebene Datei zu schreiben.

```
A>PUT CONSOLE OUTPUT TO CONSOLE
A>PUT CONSOLE CONSOLE
```

Diese Befehle weisen das Betriebssystem an, die Übertragung der Bildschirmausgabe in die Datei zu beenden. Der Computer verhält sich anschließend wieder normal.

```
A>PUT PRINTER OUTPUT TO FILE d:
filename.typ
A>PUT PRINTER FILE d:
filename.typ
```

Der Computer schreibt jetzt alle Druckerausgaben in die angeführte Datei. Zurückgesetzt wird die Druckerausgabe mit

```
A>PUT PRINTER OUTPUT TO PRINTER
A>PUT PRINTER PRINTER
```

Wie bei GET gibt es auch hier Befehle, die in eckigen Klammern hinter dem Befehlsaufruf stehen und die Datenübertragung beeinflussen.

Mit »FILTER« können Sie PUT.COM anweisen, die Control-Codes in der Zieldatei zu unterdrücken. Soll beispielsweise der ASCII-Code 3 ausgegeben werden, wandelt »FILTER« diesen in die beiden Buchstaben »^« und »C« um. In der Zieldatei steht dann »^C«. Besonders bei Ausgabe von Wordstar-Dateien auf dem Drucker ist »FILTER« nützlich. Das Gegenstück zu »FILTER« ist »NO FILTER« (Standardvorgabe).

»SYSTEM« veranlaßt, daß die Ausgaben des Betriebssystems und der Anwenderprogramme in die Zieldatei geschrieben werden, bis die Anweisung mit » PUT CONSOLE CONSOLE« widerrufen wird.

PUT.COM arbeitet aber leider nicht fehlerfrei. Angenommen Sie wollen ein sortiertes Directory der Systemdiskette 1 in eine Diskettendatei übertragen. Damit die erzeugte Datei nicht noch zusätzlich im gespeicherten Directory auftaucht, soll es auf das B-Laufwerk geschrieben werden.

```
A>PUT CONSOLE OUTPUT TO FILE B:
XYZ Putting Console Output to
File B:XYZ
A>DIR[FULL]
A>PUT CONSOLE OUTPUT TO CONSOLE
A>TYPE B:XYZ
```

Arbeiten Sie mit zwei Diskettenstationen, gibt es keine Probleme. Benutzen Sie aber den Simulator des CPC 6128 für das B-Laufwerk (und wechseln entsprechend oft die Disketten), kann es passieren, daß die Datei nur einen Teil des Directory enthält. Das muß jedoch nicht immer der Fall sein.

Wahrscheinlich steckt der Fehler tief im BIOS. Wer hiervon mehr weiß, der schreibe doch bitte an die Redaktion.

(Martin Kotulla/hg)

Über Umwege

Bildschirmausgabe, Disketten- und Dateifunktionen sind die Merkmale von »SETDEF.COM«.

Haben Sie sich auch schon über die Aufforderung »Press RETURN to Continue« geärgert, die jedesmal auftaucht, sobald eine Bildschirmseite vollgeschrieben ist. Dabei sind solche Meldungen ganz einfach zu unterdrücken. Dazu hält das Programm SETDEF.COM die Funktionen [NOPAGE] oder [NO PAGE] bereit.

```
A>SETDEF [NOPAGE]
Console Page Mode - Off
```

Die gegenteilige Wirkung bewirkt [PAGE], das den »Seiten-Modus« wieder aktiviert.

```
A>SETDEF [PAGE]
Console Page Mode - On
```

Eine weitere interessante, normalerweise nicht eingeschaltete Routine ist der »System Display Mode«. Diese Routine zeigt beim Laden eines Programms den vollständigen Dateinamen nochmals an. Eingeschaltet wird sie mit [DISPLAY] und ausgeschaltet mit [NO DISPLAY].

```
A>SETDEF [DISPLAY]
Program Name Display - On
A>pip
A:PIP      COM
CP/M 3 PIP VERSION 3.0
*^C
```

Willkommen ist diese Routine immer dann, wenn eine COM-Datei mit SYS-Attribut, die im Userbereich 0 steht, von einem anderen Benutzerbereich aus geladen werden soll. Dann enthält die Kennzeile nämlich noch die Zusatzinformation »User 0«:

```
A>SET PIP.COM [SYS]
A:PIP.COM set to directory (DIR),
Read Write (RW)
A>SETDEF [DISPLAY]
Program Name Display - On
A>USER 9
9A>pip
A:PIP      COM  (User 0)
CP/M 3 PIP VERSION 3.0
*^C
```

Alle anderen Funktionen von SETDEF.COM befassen sich direkt mit Disketten und Dateien.

Eine Funktion definiert das Laufwerk, das für temporäre Dateien vorgesehen ist. Dabei handelt es sich um Zwischendateien, die nur kurzzeitig angelegt werden. Das sind die Dateien, die meist die Kennung ».$$$« tragen. Normalerweise unsichtbar, werden sie beim nächsten Warmstart sofort wieder gelöscht. Aber nach einem Systemabsturz legt ein DIR-Befehl sie offen.

In der Regel werden sie immer auf dem aktuellen Laufwerk angelegt. Aktuell ist das Laufwerk, das im CCP-Prompt angegeben ist, beim Schneider also entweder A oder B. Ist der Platz auf dieser Diskette aber bereits ausgeschöpft, kann man den Zugriff mit SETDEF auf ein anderes Laufwerk legen. Eine RAM-Disk kann auch dazu dienen. Der Aufruf lautet [TEMPORARY=d:]. Sollen die Dateien beispielsweise auf B abgelegt werden, schreiben Sie

```
A>SETDEF [TEMPORARY=B:]
Temporary Drive - B:
```

Um wieder zum angemeldeten (aktuellen) Laufwerk für die temporären Dateien zu gelangen, ersetzt ein Stern den Laufwerksnamen.

```
A>SETDEF [TEMPORARY=*]
Temporary Drive - Default
```

Ebenso wie der nächste Befehl ergibt die Anwendung »Temporary« nur bei mindestens zwei Laufwerken einen Sinn. Von der Simulation des Zweitlaufwerks auf dem CPC 6128 raten wir in diesem Zusammenhang aber ab, da Sie sonst 90 Prozent Ihrer Zeit für das Wechseln der Disketten vergeuden.

Mit SETDEF.COM können Sie einen sogenannten »Suchpfad« (»Search Patch«) vereinbaren. Anhand dieses »Pfades« werden ».COM«- und ».SUB«-Dateien gesucht. Haben Sie beispielsweise Ihre CP/M-Systemdiskette mit den Hilfsprogrammen im Laufwerk B, eine Datendiskette in A und benutzen A als aktuelles Laufwerk, so müssen Sie beim Aufruf von COM-Dateien dem Dateinamen immer »B:« voranstellen. Das ersparen Sie sich, wenn Sie den Suchpfad gleich auf B legen.

```
A>SETDEF B:
Drive Search Path:
1st Drive - B:
```

Wollen Sie viele Hilfsprogramme im direkten Zugriff behalten, legen Sie üblicherweise die Systemdiskette in Laufwerk A und eine andere in B. Meist suchen Sie dann die gewünschte Datei auf dem falschen Laufwerk. Mit SETDEF können Sie aber auch größere Suchpfade definieren. Zum Beispiel sagen Sie dem Computer: Suche alle COM- und SUB-Dateien zuerst in A. Wenn du sie dort nicht findest, frage das Laufwerk B ab. Das sieht dann so aus:

```
A>SETDEF A:,B:
Drive Search Path
1st Drive - A:
2nd Drive - B:
```

Der Stern »*« symbolisiert wieder das aktuelle Laufwerk. Arbeiten Sie mit einer RAM-Disk als Laufwerk C, empfiehlt sich folgende Zuordnung.

```
C>SETDEF *,A:,B:
Drive Search Path:
1st Drive - Default
2nd Drive - A:
3rd Drive - B:
```

Bis zu vier Laufwerke können in den Suchpfad eingegeben werden. Keine echte Beschränkung also bei den maximal zwei Diskettenlaufwerken, die das BIOS verwalten kann.

Zum Schluß noch einmal die schon bei SUBMIT.COM angesprochene Suchreihenfolge für Dateitypen. Sie gilt auch bei SETDEF.COM. Anders als unter CP/M 2.2 darf bei CP/M 3.0 auch bei ».COM«-Dateien ausdrücklich die Extension ».COM« angegeben werden. So sieht der folgende Aufruf korrekt durchgeführt aus:

```
A>SHOW.COM
A: RW, Space: 11k
```

Daß Submit-Dateien mit der Extension >.SUB« gestartet werden dürfen, wissen Sie bereits aus dem Artikel über SUBMIT.COM in diesem Heft. Wichtig sind diese Kenntnisse für die Definition des Suchpfades nach Extensions. Denn standardmäßig gilt unter CP/M Plus folgender Suchpfad als vereinbart:

```
A>SETDEF [ORDER=COM]
Search Order - COM
```

Wollen Sie die Submit-Dateien ohne, dafür die COM-Dateien mit Extension starten, geben Sie

```
A>SETDEF [ORDER=SUB]
```

ein. Und da liegt die Tücke Ihrer CP/M-Version. Denn dieses Kommando unterstützt die gegenwärtige CP/M Plus-Version leider nicht.

```
Invalid ORDER specification
```

So müssen Sie dann auf

```
A>SETDEF [ORDER=(SUB,COM)]
Search Order - SUB, COM
```

oder

```
A>SETDEF [ORDER=(COM,SUB)]
Search Order - COM, SUB
```

ausweichen.

Bekommt der Computer einen Dateinamen ohne Extension zur Ausführung, sucht er im ersten Fall zuerst nach der ».SUB«-Datei und dann nach der ».COM«-Datei, im zweiten Fall hingegen verhält es sich umgekehrt.

Rufen Sie SETDEF.COM ohne Parameter auf, so werden die aktuellen Suchpfade und Zuweisungen angezeigt.

(Martin Kotulla/hg)

```
A>SETDEF

Drive Search Path:
1st Drive                 - A:
2nd Drive                 - B:
3rd Drive                 - C:
4th Drive                 - Default

Search Order              - COM, SUB
Temporary Drive           - Default
Console Page Mode         - On
Program Name Display      - Off
```

SETDEF.COM über sich selbst

Sekundengenau

Nutzen Sie die eingebaute Uhr, um Dateien und Programme zu kennzeichnen. Dann wissen Sie immer, welche Daten Sie wann bearbeitet haben.

Der Schneider besitzt eine eingebaute Uhr, genauer gesagt sogar vier unabhängige Zeitgeber. Doch leider ist keiner von ihnen akku- oder batteriegepuffert. So muß die Uhr nach jedem Einschalten neu gestellt werden. Da die Timer (Zeitgeber) interruptgesteuert (also softwaremäßig) arbeiten, stoppt die Uhr bei allen Diskettenzugriffen und Druckerausgaben sowie in Maschinencode-Programmen nach dem Befehl »DI« (Disable Interrupts).

Leidlich genau geht sie aber trotzdem. Mit dem Hilfsprogramm DATE.COM können sofort nach dem Systemstart die Uhrzeit und das Datum eingegeben werden.

Dazu rufen Sie das Programm mit

```
A>DATE SET
Enter today's date (MM/DD/YY):
05/01/86
Enter the time (HH:MM:SS):
16:35:00
Press any key to set time
```

auf. Beachten Sie bitte die amerikanische Schreibweise des Datums: Zuerst kommt der Monat und dann der Tag.

Da »DATE SET« auf einen Tastendruck wartet, bevor die Uhr zu laufen beginnt, können Sie die Zeit sekundengenau einstellen.

Wenn Sie die Zeit und das Datum regelmäßig benötigen, sollten Sie den Befehl »DATE SET« in die Datei PROFILE.SUB aufnehmen, so daß Sie beim Systemstart gezwungen sind, die Zeit und das Datum einzugeben.

Eine Kurzform der Eingabe ist ebenfalls möglich. Bei dieser schreiben Sie Datum und Uhrzeit direkt in die Befehlszeile hinter den Programmnamen »DATE«:

```
A>DATE 05/01/86 16:35:00
Strike key to set time
```

Der Aufruf von »DATE« ohne Parameter hat die Ausgabe des Datums und der Uhrzeit zur Folge:

```
A>DATE
Thu 05/01/86 16:37:50
```

Der Wochentag wird vom Computer automatisch berechnet und entspricht den amerikanischen Abkürzungen:

Mon – Montag
Tue – Dienstag
Wed – Mittwoch
Thu – Donnerstag
Fri – Freitag
Sat – Samstag
Sun – Sonntag

Eine Digitaluhr können Sie mit »DATE CONTINUOUS« auf dem Bildschirm darstellen.

```
A>DATE CONTINUOUS
Wed 05/07/86 16:44:30
Wed 05/07/86 16:44:31
Wed 05/07/86 16:45:32
```

Durch Drücken einer beliebigen Taste kehrt der Computer wieder zum CCP-Prompt A> (oder B>) zurück. Sie dürfen »Set« mit »S« und »Continuous« mit »C« abkürzen.

Rufen Sie mit DATE sofort nach dem Systemstart die Uhrzeit ab; ohne vorher die Zeit und das Datum eingegeben zu haben, wird stets der 15.12.1982, 00:00:00 Uhr angegeben.

CP/M Plus unterstützt die Verwaltung von Uhrzeit- und Datumseinträgen bei Dateien. So schreibt das Betriebssystem auf Wunsch bei jedem Dateizugriff das Datum und die Uhrzeit ins Directory. Über DIR[FULL] erhalten Sie dann ein sehr informatives Inhaltsverzeichnis, aus dem Sie ersehen können, wann eine Datei erzeugt wurde und zu welchem Zeitpunkt Sie das letzte Mal auf welche Datei zugegriffen haben.

Allerdings gibt es zwei Einschränkungen. Erstens können Disketten, die auf die Zeitkennung vorbereitet sind, keinesfalls mehr unter Amsdos oder CP/M 2.2 benutzt werden, weil beide Betriebssysteme mit den zusätzlichen Einträgen nichts anfangen können. Zweitens reduziert sich die Zahl der vorhandenen Directory-Einträge. Für jeweils drei Dateien wird ein zusätzlicher Eintrag im Inhaltsverzeichnis reserviert.

Wenn Sie sich trotz dieser beiden negativen Auswirkungen für die Kennzeichnung entscheiden, belohnt Sie ein bedeutend leistungsfähigeres Betriebssystem.

```
Scanning Directory...
Sorting Directory...
Directory For Drive A: User 0
```

Name	Bytes	Recs	Attributes	Prot	Update	Access
AMSDOS	COM	1k	8 Dir RW	None		
BANKMAN	BAS	1k	7 Dir RW	None		
BANKMAN	BIN	2k	12 Dir RW	None		
C10CPM3	EMS	25k	200 Dir RW	None		
DATE	COM	3k	23 Dir RW	None		04/04/86 00:32
DEVICE	COM	8k	58 Dir RW	None		04/03/86 23:10
DIR	COM	15k	114 Dir RW	None		04/04/86 00:35
DISCKIT3	COM	6k	48 Dir RW	None		04/03/86 23:26
ED	COM	10k	73 Dir RW	None		04/03/86 23:26
ERASE	COM	4k	29 Dir RW	None		04/03/86 23:09
GET	COM	7k	51 Dir RW	None		04/03/86 23:26
KEYS	CCP	1k	3 Dir RW	None		04/04/86 00:32
KEYS	WP	1k	3 Dir RW	None		04/03/86 22:53
LANGUAGE	COM	1k	8 Dir RW	None		04/04/86 00:33
PALETTE	COM	1K	8 Dir RW	None		04/03/86 23:09
PIP	COM	9k	68 Dir RW	None		04/03/86 23:26
PROFILE	SUB	1k	1 Dir RW	None		04/04/86 00:32
PUT	COM	7k	55 Dir RW	None		04/04/86 00:35
RENAME	COM	3k	23 Dir RW	None		04/03/86 23:09
SET	COM	11k	81 Dir RW	None		04/03/86 23:10
SET24x80	COM	1k	8 Dir RW	None		04/04/86 00:34
SETDEF	COM	4k	32 Dir RW	None		04/04/86 00:34
SETKEYS	COM	2k	16 Dir RW	None		04/04/86 00:33
SETLST	COM	2k	16 Dir RW	None		04/04/86 00:34
SETSIO	COM	2k	16 Dir RW	None		04/03/86 23:09
SHOW	COM	9k	66 Dir RW	None		04/03/86 23:26
SUBMIT	COM	6k	42 Dir RW	None		04/04/86 00:33
TYPE	COM	3k	24 Dir RW	None		04/03/86 22:52

```
Total Bytes     = 146k Total Records = 1093 Files Found =   28
Total 1k Blocks = 146  Used/Max Dir Entries For Drive A: 64/28
```

Das Aussehen des Inhaltsverzeichnisses hat sich verändert

Dazu muß eine Diskette ausdrücklich für das »Time- and Date-Stamping« vorbereitet werden. Diese Aufgabe übernimmt das Programm INITDIR.COM. Es wird zusammen mit der Bezeichnung des Laufwerks aufgerufen, in dem sich die umzuformatierende Diskette befinden muß.

```
A>INITDIR A:
INITDIR will activate time
stamps for specified drive.
Do you want to re-format the
directory on drive: A (Y/N)? Y
A>
```

Mit einem erneuten Aufruf von INITDIR bei einer bereits umformatierten Diskette kann das Datum wieder entfernt werden.

```
A>INITDIR A:
INITDIR will activate time stamps
for specified drive.
Do you want to re-format the
directory on drive: A (Y/N)? Y
Directory already re-formatted.
Do you want to recover time/date
directory space (Y/N)? Y
```

Nach dieser Prozedur ist der belegte Platz im Directory wieder frei.

Nehmen wir aber an, Sie wollen die Kennung beibehalten. Dazu muß diese mit SET.COM aktiviert und in den gewünschten Modus gebracht werden.

Es gibt drei verschiedene Modi: »CREATE«, »UPDATE« und »ACCESS«. »CREATE« und »ACCESS« schließen einander aus.

A>SET [CREATE=ON] schreibt dem Computer vor, beim Einrichten einer Datei die Zeit und das Datum zu notieren.

A>SET [ACCESS=ON] erzeugt Zeit- und Datumseinträge bei jedem Zugriff auf eine Datei.

A>SET [UPDATE=ON] trägt das Datum und die Uhrzeit der jeweils letzten Änderung einer Datei ins Directory ein.

Nach dem SET-Kommando hat sich das Aussehen des Inhaltsverzeichnisses bei DIR[FULL] erheblich verändert. Das Bild zeigt, wie das Directory der Systemdiskette 1 jetzt aufgebaut ist.

(Martin Kotulla/hg)

CP/M-Kontraste

»HELP.COM« und »ED.COM« - Programme, die aus zwei total verschiedenen CP/M-Epochen stammen. Nirgendwo werden die Kontraste innerhalb CP/M Plus so deutlich wie hier.

CP/M Plus ist ein Betriebssystem mit Widersprüchen. Auf der einen Seite erhebt es den Anspruch, ein modernes und professionell einsetzbares System zu sein. Auf der anderen Seite besitzt es aber Eigenheiten, die noch aus den allerersten Versionen von CP/M herrühren, und keinesfalls mehr der heutigen Zeit entsprechen.

Ein besonders ärgerliches Fossil aus »alten« Zeiten ist der Editor »ED.COM«. Die CP/M-Fan-Gemeinde wünscht sich schon seit Jahren, daß man ihn durch einen bildschirmorientierten Editor ersetzt. Daß Digital Research mit dieser Sparpolitik aber nicht allein auf weiter (Computer-)Flur steht, hat Microsoft mit dem MS-DOS-Editor »EDLIN« bewiesen, der genauso umständlich zu bedienen ist.

Viele Benutzer von CP/M sind deshalb schon seit langem auf andere Editoren umgestiegen. So bietet sich beispielsweise Wordstar (unter dem Menüpunkt »N«) an, das eine sehr komfortable Programmbearbeitung erlaubt. Wer Turbo-Pascal besitzt, der findet dort auch einen sehr komfortablen Editor.

Es erübrigt sich deshalb, den Zeileneditor ED.COM ausführlich zu erklären. Wenn Sie aber gewisse selbstquälerische Züge an sich entdecken und ihn trotzdem benutzen wollen, so finden Sie im Bild 1 sämtliche Befehle von ED.COM. Der Aufruf ist einfach.

```
A>ED DATEI
```

Ein repräsentativeres Aushängeschild für ein professionelles Betriebssystem stellt das Programm »HELP.COM« dar. Sie finden es auf der dritten Seite der Systemdisketten Ihres CPC 6128. Dieses Programm ersetzt fast ein Handbuch. Über das hierarchisch aufgebaute System erhält man Informationen über die Funktionen und die Bedienung der einzelnen CP/M-Dienstprogramme.

HELP.COM startet durch Eingabe von »HELP«.

```
A>HELP
```

Die Bildschirmausgabe ändert sich auf das in Bild 2 gezeigte Format.

HELP.COM benutzt ein eigenes Aufforderungszeichen (Prompt).

```
HELP>
```

Angenommen Sie wissen nicht mehr, wie DATE.COM funktioniert und das Handbuch ist an den Freund ausgeliehen - also nicht greifbar.

Es interessiert Sie aber brennend, was sich mit DATE.COM alles anstellen läßt. Dazu laden Sie HELP.COM, warten auf die Eingabeaufforderung und tippen:

```
HELP> DATE
```

Das Diskettenlaufwerk läuft kurz an, und der Computer lädt die Datei mit den passenden Hilfstexten nach.

```
DATE

Syntax:

      DATE [CONTINUOUS]
      DATE [time-specification]
      DATE SET

Explanation:
The DATE command lets you display
and set the date and time of day.
ENTER .subtopic FOR INFORMATION
ON THE FOLLOWING SUBTOPICS:
EXAMPLES

HELP>
```

Nun sind Sie umfassend informiert. Die Befehlssyntax ist klar. DATE.COM kann laut Angaben von HELP.COM mit

```
A>DATE
A>DATE CONTINUOUS
A>DATE mm/tt/jj ss:mm:ss
A>DATE SET
```

aufgerufen werden. Alle Texte, die bei der Syntaxdefinition in Klammern angegeben sind, sind optional und dürfen auch weggelassen werden. Optionen bei anderen Befehlen, die ein senkrechter Strich (»|«) trennt, dürfen wahlweise benutzt werden. Solche Optionen schließen sich gegenseitig aus - beispielsweise [PAGE |NOPAGE].

Unter »Explanation« finden Sie stets eine - mehr oder weniger ausführliche - englischsprachige Beschreibung dessen, was die Routine leistet.

Oft helfen solche Angaben allerdings nicht sehr viel weiter. Man will schließlich Beispiele oder eine genauere Beschreibung abrufen. Dazu sind die »Subtopics« gedacht - sozusagen Unterüberschriften. Wenn nach den Informationen über DATE.COM wieder der HELP>-Prompt erscheint, finden Sie darin eine Liste der Unterthemen. Bei DATE.COM fällt nur ein einziges Thema an - nämlich »Examples«. Geben Sie dann das Wort »EXAMPLES« mit einem vorangestellten Punkt ein.

```
HELP> .EXAMPLES
```

nA	Hole n Zeilen aus der Datei in den RAM-Speicher.
0A	Hole Zeilen aus der Datei, bis der RAM-Speicher halbvoll ist.
#A	Hole Zeilen aus der Datei, bis der RAM-Speicher voll ist.
B	Setze Cursor an den Anfang des Textes im RAM-Speicher.
-B	Setze Cursor an das Ende des Textes im RAM-Speicher.
nC	Bewege den Cursor n Zeichen nach unten.
-nC	Bewege den Cursor n Zeichen nach oben.
nD	Lösche n Zeichen ab der Cursorposition.
-nD	Lösche n Zeichen vor dem Cursor.
E	Speichere Datei ab und kehre in den CCP zurück.
Ftext CTRL-Z	Suche die Zeichenfolge »text«.
H	Speichere die Datei im RAM und nehme Editieren wieder auf.
I	Gehe in den Texteinfügemodus (wird durch CTRL-Z beendet).
Itext CTRL-Z	Füge die Zeichenkette »text« an der Cursorposition ein.
Jsearch string^Zinsert	Nach dem »search string« wird der »insert string« eingesetzt, alle Zeichen bis zum »delete-to string« werden gelöscht.
nK	Lösche n Zeilen nach der Cursorposition.
-nK	Lösche n Zeilen vor dem Cursor.
nL	Bewege den Cursor n Zeilen nach unten.
-nL	Bewege den Cursor n Zeilen nach oben.
nMcommand	Wiederhole das folgende Kommando n Mal.
n	Bewege den Cursor n Zeilen nach unten.
-n	Bewege den Cursor n Zeilen nach oben.
n:	Bewege den Cursor in die Zeile n.
:ncommand	Führe ein Kommando bis zur Zeile n aus.
Ntext CTRL-Z	Erweiterte Suche nach der Zeichenkette »text«.
O	Mache alle Änderungen an dem Text im RAM rückgängig.
nP	Bewege den Cursor n Zeilen nach unten und zeige diese an.
-nP	Bewege den Cursor n Zeilen zurück und zeige diese an.
Q	Breche Bearbeitung ab, ohne die Datei zu speichern.
R	Lese die Zwischendatei X$$$$$$$.LIB in den RAM-Speicher.
Rtext	Lese die Datei mit dem Namen »text« in den Speicher.
Soldstring CTRL-Z newstring CTRL-Z	Ersetze »oldstring« durch »newstring«.
nT	Gebe n Zeilen unterhalb der Cursorposition aus.
-nT	Gebe n Zeilen oberhalb der Cursorposition aus.
U	Wandle in Großbuchstaben um (vorwärts).
-U	Wandle in Großbuchstaben um (rückwärts).
V	Schalte Zeilennumerierung ein.
-V	Schalte Zeilennumerierung aus.
0V	Zeige den freien Speicherplatz im RAM.
nW	Schreibe n Zeilen in die neue Datei.
nX	Schreibe/füge n Zeilen an X$$$$$$$.LIB an.
nXtext	Schreibe n Zeilen in die Datei mit dem Namen »text«.
0X	Lösche die Zwischendatei X$$$$$$$.LIB.
0Xtext	Lösche die Datei mit dem Namen »text«.
nZ	Warte n Sekunden bis zur Ausführung des nächsten Befehls.

Bild 1. Hier finden Sie den gesamten Befehlssatz des Programms ED.COM

```
A>help

HELP UTILITY V1.1

At "HELP>" enter topic {subtopic}...

EXAMPLE:  HELP> DIR BUILT-IN

Topics available:

COMMANDS         CNTRLCHARS        COPYSYS
DATE             DEVICE            DIR
DISCKIT3         DUMP              ED
ERASE            FILESPEC          GENCOM
GET              GSX               HELP
HEXCOM           INITDIR           LANGUAGE
LIB              LINK              MAC
PALETTE          PATCH             PIP (COPY)
PUT              RENAME            RMAC
SAVE             SET               SET24X80
SETDEF           SETKEYS           SETLST
SETSIO           SHOW              SID
SUBMIT           TYPE              USER
XREF

HELP>
```

Bild 2. Hilfe vom System – die Help-Utility

Der Computer gibt dann folgenden Text aus.

```
DATE
  EXAMPLES

A>DATE
Displays the current date
and time.

A>DATE C
Display the date
and time continuously.

A>DATE 08/14/82 10:30:0
Sets the date and time.

A>DATE SET
Prompts for date
and time entries.

HELP>
```

Manche Programmbeschreibungen enthalten weitere Subtopics. Sie werden ebenfalls mit einem vorangestellten Punkt aufgerufen.

Wollen Sie wieder ins Hauptmenü zurückkehren, das alle vorhandenen Programmbeschreibungen auflistet, drücken Sie beim Erscheinen des HELP>-Zeichens »?«.

```
HELP> ?
```

Es erscheint folgende Liste:

```
Topics available

COMMANDS CNTRLCHARS COPYSYS DATE
DISCKIT3 DUMP       ED      ERASE
GET      GSX        HELP    HEXCOM
DEVICE   DIR
FILESPEC GENCOM
INITDIR  ...
```

Erinnern Sie sich noch an PIP.COM und den schönen Begriff »Multi Command Mode«? Das ist genau das, womit Sie bis jetzt bei HELP.COM gearbeitet haben. Sie können aber auch schon in der Aufrufzeile die gewünschten Themengebiete spezifizieren.

```
A>HELP DIR
A>HELP HEXCOM
A>HELP GSX
```

Auch die Subtopics lassen sich direkt angeben.

```
A>HELP DIR BUILT-IN
```

Eine Stufe in der Hierarchie nach unten, und die Sache sieht so aus:

```
A>HELP DIR BUILT-IN EXAMPLES
```

Bis zu neun hierarchische Stufen verarbeitet HELP.COM, die Hauptstufe und acht Subtopics. Dafür, daß Sie bei HELP immer das Richtige in ausreichendem Umfang finden, sorgt die Datei HELP.HLP mit einer Länge von 75 KByte.

HELP.COM sieht zwei Funktionen vor: [NOPAGE] und [LIST]. Während Sie die Erklärung von »NOPAGE« schon von den anderen in diesem Heft beschriebenen Programmen her kennen, tritt »LIST« hier zum ersten Mal auf. Es dient dazu, bei der Druckerausgabe mit CTRL-P Papier zu sparen. Es werden nämlich einige der zahlreichen Leerzeilen aus dem Ausdruck entfernt.

HELP.COM besitzt aber noch eine sehr nützliche Eigenschaft. Sie können

```
.
.
.
///2PROFILE.SUB
Everytime you power up or reset your computer, CP/M Plus looks for a special SUBmit file named PROFILE.SUB to execute. If it does not exist, CP/M Plus resumes normal operation. If the PROFILE.SUB file exists, the system executes the commands in the file. This file is convenient to use if you regularly execute a set of commands before you do your regular session on the computer.
///1TURBO
Syntax:    TURBO
Funktion: Turbo-Pascal ist einer der erfolgreichsten Pascal-Compiler mit einer integrierten Editor-, Compiler- und Laufzeitumgebung.
///2BEISPIELE
A>TURBO
Include error messages? Y
Bindet die Fehlermeldungen in den Compiler ein. Dadurch sinkt der verfügbare Speicherplatz um ca. zwei KByte.
Include error messages? N
Weist Turbo-Pascal an, TURBO.MSG aus dem Compiler herauszuhalten und ist bei knappem Speicherplatz zu empfehlen.
///2PRODUKTE
Die Herstellerfirma bietet eine ganze Reihe von Erweiterungen für Turbo-Pascal an. Auch andere Softwarefirmen versuchen, sich an den Erfolg von Turbo-Pascal mit Programmsammlungen und "Toolkits" anzuhängen. Beachten Sie, daß viele der Programme nur unter MS-DOS laufen.
///3BEISPIELE
Turbo Tutor, Turbo Toolbox, Turbo Graphix Toolbox, Turbo Lader, Turbo Database, Turbo Gameworks, Turbo Editor
///1TYPE
Syntax:  TYPE {filespec {[PAGE | NOPAGE ]}}
.
.
.
```

Bild 3. Informationen für Turbo-Pascal

```
A>HELP TURBO
A>HELP TURBO BEISPIELE
A>HELP TURBO PRODUKTE
A>HELP TURBO PRODUKTE BEISPIELE
```

Bild 4. So ruft man die Daten für Turbo-Pascal in HELP.COM ab .

es mit eigenen Texten erweitern, und somit in das Programm eigene Texte und eigene Beschreibungen aufnehmen oder auch die englischen Texte ins Deutsche übersetzen.

Die in der Datei HELP.HLP stehenden Texte sind speziell codiert und lassen sich deshalb nicht ohne weiteres bearbeiten. Sie müssen HELP.COM erst ausdrücklich anweisen, die Datei in Klartext zu verwandeln. Dazu benutzen Sie den Befehl [EXTRACT].

HELP.HLP hat, wie schon erwähnt, eine Länge von 75 KByte. Mit »EXTRACT« erzeugt der Computer eine Datei HELP.DAT mit wiederum 74 KByte. Dies bedeutet, daß Sie außer HELP.COM und HELP.HLP alle anderen Dateien und Programme von der Diskette löschen müssen. Denn sonst steht nicht genug Platz zur Verfügung, um die neue Datei anzulegen. Nach der Löschaktion verbleiben noch freie 96 KByte auf der Diskette.

Weisen Sie nun den Computer an, die Datei umzuschichten:

```
A>HELP [EXTRACT]
Extracting data....
```

Gedulden Sie sich aber dabei etwas, denn es ist für den Computer kein Kinderspiel, mit so einer Riesendatei zu hantieren. Das Ende erkennen Sie an

```
HELP.DAT created
```

Und »DIR« zeigt an, daß eine neue Datei im Inhaltsverzeichnis steht.

```
A>DIR
A: HELP  DAT : HELP  COM :
HELP  HLP
```

Mit »TYPE HELP.DAT« und eventueller Druckerausgabe können Sie sich nun diese Datei ansehen und den Aufbau studieren.

Mit Hilfe eines Textprogramms wird HELP.DAT editiert. Nach der Änderung ist die Datei HELP.DAT wieder zu codieren, was mit Hilfe von [CREATE] geschieht.

```
A>HELP [CREATE]
Creating HELP.HLP....
```

Auch hier dauert es wieder einige Zeit, bis der Computer die Datei HELP.HLP fertiggestellt hat und folgende Meldung erscheint:

```
HELP.HLP created
```

Sie können nun HELP.COM starten und sich die veränderten Texte anschauen.

Falls sich unerklärlicherweise in die Texte Grafikzeichen eingeschlichen haben, die Sie niemals eingegeben haben, benutzt der Editor das siebte Bit der Zeichen zur Kennzeichnung bestimmter Formatierungen. Sie müssen dann diese Bits mit Hilfe von PIP.COM mit der Z(ero)-Option entfernen.

```
A>PIP HELP2=HELP.DAT[Z]
A>ERA HELP.DAT
A>REN HELP.DAT=HELP2
```

Schwieriger wird es, wenn Sie HELP.COM mit eigenen Texten erweitern wollen. Dazu gibt es nämlich einige Vorschriften zu beachten.

Jede Überschrift eines Befehls oder eines Subtopics leiten drei Divisionsstriche »///« ein. Darauf folgt eine Zahl, die die gewünschte hierarchische Ebene angibt. »1« steht für die Hauptebene, die Werte »2« bis »9« für die verschiedenen Unterebenen. An diese Zahl hängen Sie den Namen des Programms oder den Titel des Untermenüs an. Es folgt ein Carriage Return.

```
///1Turbo-Pascal<CR>
```

Die Hauptmenüs müssen alphabetisch geordnet sein. Also kommt »Turbo-Pascal« vor »Wordstar«, und »dBASE II« hinter »CBASIC«.

Die Unterbegriffe sollen ebenfalls innerhalb einer Hauptgruppe eine alphabetische Reihenfolge einhalten. »Optionen« steht demnach hinter »Beispiele«, und zwischen beide wird die Rubrik »Fehlermeldungen« angelegt.

Um beispielsweise »Turbo-Pascal« zu beschreiben, hat sich die Anordnung aus Bild 3 bewährt.

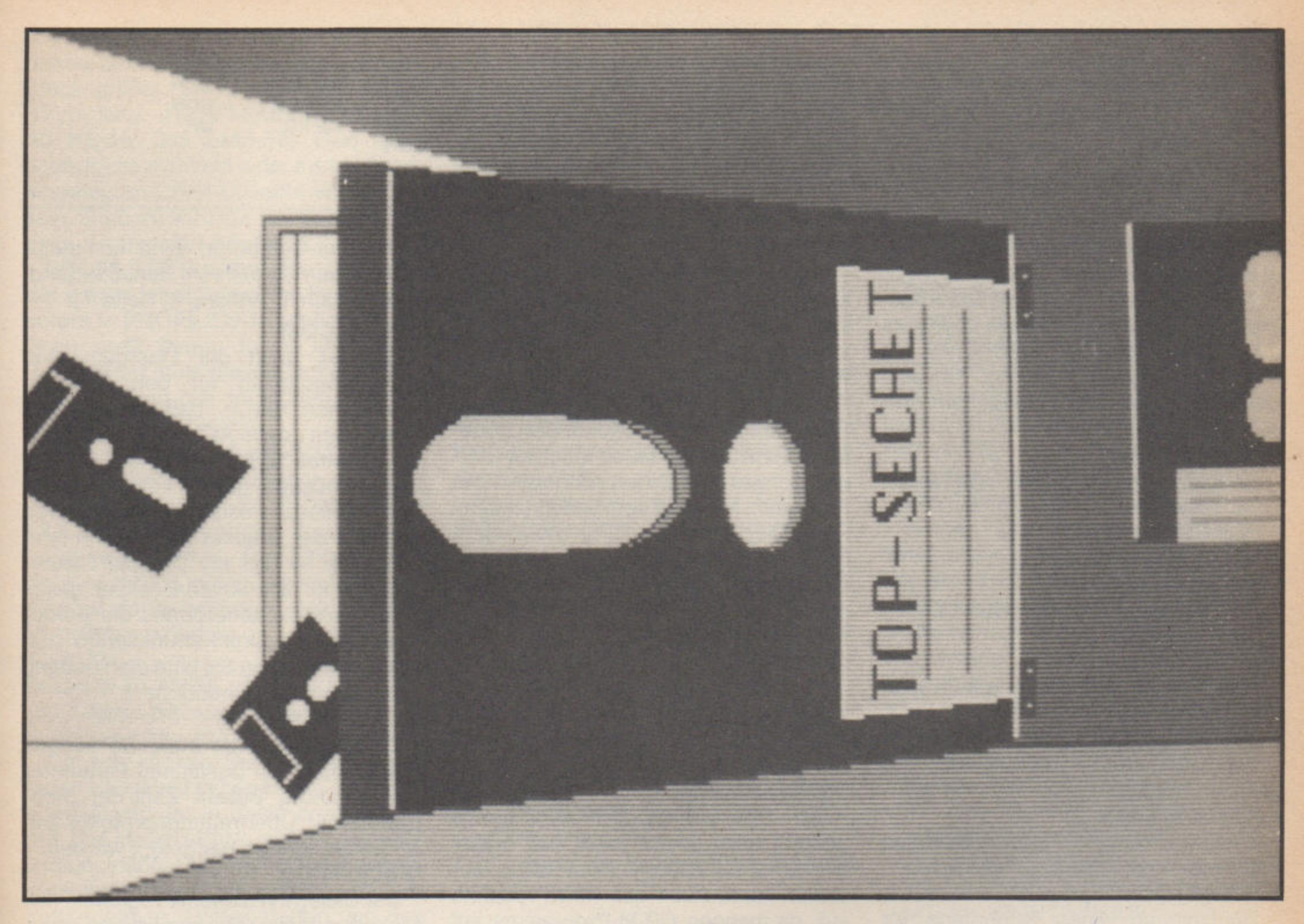
Auch diese Anpassung von HELP.COM bindet [CREATE] in die HELP-Utility ein. Bild 4 zeigt, wie man die Daten über Turbo-Pascal aufruft.

```
A>HELP [CREATE]
```

Das läßt wieder einmal spüren, wie angenehm man mit CP/M Plus arbeiten kann und wie weit Digital Research dieses Betriebssystem für zukünftige Verbesserungen vorbereitet hat.

(Martin Kotulla/hg)





CP/M Plus für Joyce

Auch der Joyce besitzt, wie der CPC 6128, das Betriebssystem CP/M Plus. Hier erfahren Sie Gemeinsamkeiten und Unterschiede der beiden Versionen.

Während beim Schneider CPC das CP/M-Betriebssystem sozusagen eine kostenlose Dreingabe ist, gehört es beim Joyce CP/M Plus zum festen Inventar und ist für diesen Computer lebenswichtig. Denn ohne CP/M laufen außer Locoscript keinerlei Programme.

Im Gegensatz zu den CPCs lädt der Joyce das Betriebssystem automatisch beim Einschalten des Computers. Je nachdem, welche Seite der Diskette eingelegt ist, startet entweder das Textverarbeitungssystem Locoscript oder das CP/M Plus-Betriebssystem. Sie können CP/M aber auch durch gleichzeitiges Drücken von »EXTRA«, »SHIFT« und »EXIT« starten – vorausgesetzt, die CP/M Plus-Systemdiskette ist richtig herum eingelegt.

Nach einigen Sekunden erscheint folgende Startmeldung:

```
CP/M Plus Amstrad Consumer
Electronics plc v 1.2, 61K TPA,
1 Laufwerk, 112K Laufwerk M:
```

Dieses Titelbild sagt schon einiges aus: Die TPA (Transient Program Area), der Bereich, in den CP/M-Programme und Daten geladen werden, hat einen Umfang von 61 KByte, und ist damit genausogroß wie beim CPC 6128. Fragen Sie sich jetzt, was man mit den übrigen 256-61=195 KByte beim PCW 8256 beziehungsweise 512-61= 451 KByte beim »Fat Joyce« anstellen kann? 112 beziehungsweise 368 KByte sind als RAM-Disk verfügbar. »RAM-Disks« sind einfach Speicherbereiche, die sich so verhalten, als wären sie eine zusätzliche Disketten-Station. Das Betriebssystem verwaltet sie so geschickt, daß jedes Anwenderprogramm glaubt, es wäre ein mechanisches Laufwerk mehr im System vorhanden.

Der Vorteil der RAM-Disk liegt in der vielfach höheren Geschwindigkeit. Diese angenehme Eigenschaft wird allerdings mit einem Nachteil erkauft. Die RAM-Disk kann nur so lange Daten speichern, wie der Computer eingeschaltet ist. Um den Inhalt dauerhaft zu sichern, müssen Sie die Daten am Ende der Arbeit auf eine Diskette kopieren. Bei einem Stromausfall sind alle Daten verloren. Die RAM-Disk hört beim Joyce auf den Namen »M«. Dieser Name steht für »Memory Disk«. Geben Sie, direkt nach dem Hochfahren des Betriebssystems (»Booten«), den Befehl:

```
A>DIR M:
```

ein. Da die RAM-Disk noch leer ist, erscheint die Meldung: No File.

Wenn Sie die Geschwindigkeit von CP/M Plus erhöhen wollen, kopieren Sie einfach alle Dateien und Programme der Systemdiskette nach M und wählen M als Default-Laufwerk:

```
A>PIP M:=*.*
A>M:
M>
```

Kleine Unterschiede

Sowohl der CPC 6128 als auch der Joyce verwenden CP/M Plus. Dennoch kann nicht alles, was für den CPC 6128 gilt, ohne weiteres auf den Joyce übertragen werden. So ist eine – übrigens äußerst angenehme – Eigenschaft von Joyce, daß der größte Teil des Benutzerdialogs in deutscher Sprache ab-

```
J12DCPM3 EMS : BASIC    COM : DIR     COM : ED      COM : ERASE   COM
LANGUAGE COM : PALETTE  COM : PAPER   COM : PIP     COM : RENAME  COM
SET      COM : SET24X80 COM : SETDEF  COM : SETKEYS COM : SETLST  COM
SETSIO   COM : SHOW     COM : SUBMIT  COM : TYPE    COM : KEYS    WP
RPED     BAS : RPED     SUB : PROFILE GER : DISCKIT COM
```

Bild. Der Inhalt der Joyce-Systemdiskette

läuft. Unser Bild zeigt das Directory der Systemdiskette (Seite 2) vom Joyce.

Betrachten wir zuerst die Standardprogramme, die sowohl beim Joyce als auch beim Schneider CPC 6128 auf Diskette mitgeliefert werden. Sie werden in anderen Artikeln in diesem Heft ausführlich besprochen.

- **DIR.COM** (Seite 2)
  Ausgabe des Inhaltsverzeichnisses einer Diskette
- **ED.COM** (Seite 2)
  Ein einfacher Editor
- **ERASE.COM** (Seite 2)
  Programm zum Löschen von Dateien auf der Diskette
- **PIP.COM** (Seite 2)
  Universelles Programm zum Kopieren von Dateien
- **RENAME.COM** (Seite 2)
  Programm zum Umbenennen von Diskettendateien
- **SET.COM** (Seite 2)
  System-Programm zum Festlegen von Datei- und Disketten-Parametern
- **SETDEF.COM** (Seite 2)
  Bestimmung eines Suchpfads für Dateien und des Laufwerks für temporäre Dateien
- **SHOW.COM** (Seite 2)
  Anzeige spezieller Disketten- und Laufwerks-Parameter
- **SUBMIT.COM** (Seite 2)
  Automatisierte Bearbeitung von Befehlen aus einer Diskettendatei
- **TYPE.COM** (Seite 2)
  Ausgabe von ASCII-Textdateien auf dem Bildschirm
- **GET.COM** (Seite 3)
  Eingabeumleitung aus einer Diskettendatei
- **PUT.COM** (Seite 3)
  Ausgabeumleitung in eine Datei auf der Diskette
- **INITDIR.COM** (Seite 3)
  Vorbereitung des Directory für Zeit- und Datumseinträge
- **DATE.COM** (Seite 3)
  Setzen und Auslesen der Uhrzeit und des Datums
- **DEVICE.COM** (Seite 3)
  Anzeige und Abänderung der Zuordnung von logischen und physikalischen Geräten
- **HELP.COM** (Seite 4)
  Hilfsprogramm, das Informationen über andere CP/M-Programme ausgibt.

Einige Programme haben ähnliche oder gleiche Namen wie ihre Pendants beim CPC 6128, dennoch fallen kleine Unterschiede auf.

LANGUAGE.COM

Dieses Programm stellt den Zeichensatz für die Bildschirmausgabe ein. Identisch mit dem 6128-Programm LANGUAGE.COM.

PALETTE.COM

PALETTE erlaubt die Änderung der Bildschirmfarben. Nachdem der Joyce-Bildschirm und der Video-Chip nur Grün und Schwarz darstellen können, ist das Programm nicht mit dem gleichnamigen PALETTE.COM des CPC 6128 identisch. Es gibt in der Joyce-Version sinnvollerweise nur zwei Farbkombinationen.

```
A>PALETTE 1 0
A>PALETTE 0 1
```

»1 0« wählt schwarze Schrift auf grünem Untergrund (Inversdarstellung), »0 1« hingegen grün auf schwarz (normale Arbeitsweise).

SET24X80.COM

Der Unterschied zum CPC 6128 besteht beim Joyce darin, daß dieser Computer 31 Zeilen je 90 Zeichen darstellen kann. Auch das wirft Probleme auf, da manche CP/M-Programme auf einen 24zeiligen Bildschirm bestehen.

```
A>SET24X80
A>SET24X80 ON
```

Beide Anweisungen sind gleichwertig und schalten den verkleinerten Bildschirm ein. Abgeschaltet wird er mit:

```
A>SET24X80 OFF
```

SETKEYS.COM

Dieses Hilfsprogramm stellt genau wie beim CPC 6128 die Tastaturbelegung ein. Hier gelten folgende Bezeichnungen:

- »N« für Normal.
- »S« für Shift.
- »A« für die ALT-Ebene.
- »E« für die EXTRA-Ebene.
- »SA« für SHIFT / ALT.

Es können mehrere Tastaturebenen auf einmal angegeben werden. Sie müssen deren Bezeichner aber mit einem Leerzeichen trennen. Schreiben Sie also nicht »NSE«, sondern »N S E«.

Erweiterungszeichen (beim Joyce können auch die Funktionstasten belegt werden) definieren Sie, indem Sie der Zeile in der Belegungsdatei ein »E« voranstellen:

```
E # 80 "DIR^M"
```

»E« steht für »Expansion Key«. Verwechseln Sie das aber bitte nicht mit dem Umbelegen der EXTRA-Taste:

```
32 E "A"
```

Die Tastaturnummern entsprechen in weiten Zügen denen der anderen Schneider-Computer. Eine komplette Liste finden Sie in Ihrem Benutzerhandbuch (Buch 1, Anhang 1, Seite 7).

SETLST.COM

SETLST dient der Übergabe von Steuersequenzen an den Drucker. Abgesehen davon, daß die Control-Taste beim Joyce ALT heißt, und damit die Control-Codes »Alternate-Codes« heißen müssen, gibt es keine weiteren Unterschiede. Positiv fällt beim Joyce auf, daß der Drucker ins System integriert ist, so daß die Drucker-Steuerzeichen für alle Joyce-Besitzer gleich sind. Welche Steuercodes die jeweils gewünschten Druckerfunktionen aufrufen, entnehmen Sie bitte dem Anhang 2 des Benutzerhandbuchs 1.

SETSIO.COM

Dieses Programm stellt die serielle Schnittstelle auf bestimmte Parameter ein. Baudrate, Bitzahl, Zahl der Stop-Bits, Parität, Übertragungsprotokoll und so weiter können festgelegt werden.

DISCKIT.COM

Der Dialog mit diesem Programm, das Disketten formatiert, kopiert und überprüft, ist vollständig in Deutsch gehalten. Als Zwischenspeicher beim Kopieren kompletter Disketten dient automatisch die RAM-Disk. Das spart Zeit und ist komfortabler als das ständige Diskettenwechseln beim 6128-Programm DISCKIT3.COM beziehungsweise seinem CP/M 2.2-Bruder DISCKIT2.COM. Ist ein zweites Laufwerk angeschlossen, erkennt DISCKIT.COM es automatisch und bietet weitere Menüs, mit denen Sie das Quell- und Ziellaufwerk festlegen können.

Wenig Joyce-spezifisches

Ansonsten gibt es nur wenige Programme und Dateien, die nicht zum CP/M Plus-Standard gehören. Zu nennen sind allenfalls BASIC.COM und PAPER.COM, sowie die nicht direkt zu den CP/M-Programmen gehörenden Dateien J12CPM3.EMS, RPED.BAS, RPED.SUB und PROFILE.GER.

Beginnen wir mit der EMS-Datei. Keiner der CP/M Plus-Computer vermag das Betriebssystem wie bei der Version 2.2 in den Systemspuren der Diskette abzulegen. Deshalb steht es in einer EMS-Datei. Heißt diese beim CPC 6128 »C10CPM3.EMS«, wird sie beim



Joyce »J12CPM3.EMS« genannt. Nebenbei bemerkt, ist letztere ein ganzes Stück größer: C10CPM3.EMS ist 25 KByte lang, J12CPM3.EMS immerhin 40 KByte. Das EMS-File muß auch beim Joyce nicht auf jeder ins Laufwerk eingelegten CP/M-Diskette vorhanden sein, sondern wird nur ein einziges Mal beim Systemstart benötigt.

PROFILE.GER sollten Sie sofort bei der ersten Inbetriebnahme Ihres Computers in PROFILE.SUB umbenennen. Dann stellt Ihnen Joyce nach dem Systemstart stets die deutsche Tastenbelegung und die deutschen Sonderzeichen zur Verfügung.

```
A>REN PROFILE.SUB=PROFILE.GER
```

»GER« steht für »German«, während die Extension der 6128-Datei PROFILE.ENG die englische Tastaturbelegung repräsentiert. Die Eingabe des Schlüsselworts »BASIC« beauftragt das CP/M-Betriebssystem, den Basic-Interpreter zu laden:

```
A>BASIC
Mallard-80 BASIC with Jetsam
(c) Copyright 1984 Locomotive
Software Ltd. All rights reserved
```

Danach können Sie den Joyce wie jeden handelsüblichen Heim- oder Personal Computer in Basic programmieren. Das Mallard-Basic ist übrigens sehr leistungsfähig und besitzt ausgezeichnete Befehle zur Verwaltung von Diskettendateien. Leider stehen von den maximal vorhandenen 512-KByte-RAM des Joyce unter Basic nur etwa 30 KByte für ein Programm zur Verfügung. Aber keine Sorge: Die RAM-Disk M kann man auch von Mallard-Basic aus ansprechen. Zusammen mit den leistungsfähigen Kommandos CHAIN und CHAIN MERGE lassen sich mit etwas Überlegung sehr umfangreiche verkettete Programmsysteme entwickeln.

Von Mallard-Basic aus gelangen Sie durch Eingabe des Befehls »SYSTEM« wieder in den Console Command Processor (CCP) des CP/M-Betriebssystems zurück.

Beim Basic-Start dürfen einige Parameter angegeben werden. So können Sie den Computer anweisen, ein Basic-Programm zu laden und sofort zu starten:

```
A>BASIC PACMAN
```

Mit »/F:n« wird die Zahl der Dateien (»Files«) angegeben, die unter Basic gleichzeitig geöffnet sein dürfen. »n« darf von 0 bis 255 alle Werte annehmen. Vorgegeben sind drei Dateipuffer. Wollen Sie zum Beispiel 10 Dateien öffnen, schreiben Sie:

```
A>BASIC /F:10
```

»/M:n« bestimmt die Obergrenze (»High Memory«) des für Mallard-Basic freigegebenen RAM-Speichers. Dieser Parameter ist nützlich, wenn Sie gleichzeitig Maschinencode-Routinen an das obere Speicherende legen wollen. Der Wert für »/M:n« muß als vorzeichenlose Integerzahl (0 bis 65535), als Hexadezimalzahl (&0000 bis &FFFF) oder als oktale Zahl (&O0000 bis &O177777) eingegeben werden. Ist keine Begrenzung angegeben, holt sich Mallard-Basic die Startadresse des BDOS aus der RAM-Adresse 6 und wertet sie als Obergrenze des Speichers. Natürlich kann für »/M:n« keine Zahl stehen, die größer ist, als es das System überhaupt gestattet.

Der letzte Parameter heißt »/S:n« und bestimmt, wie viele frei adressierbare Sätze maximal verwendet werden dürfen. Ohne den S-Parameter liegt dieser Wert bei 128.

Der bildschirmorientierte Texteditor RPED.BAS zählt ebenfalls zu den Basic-Programmen. Er verwaltet einen Textpuffer von etwa 20000 Zeichen und kann direkt von der CCP-Kommandoebene des Betriebssystems mit Hilfe der Datei RPED.SUB gestartet werden:

```
A>RPED.SUB
```

Oder:

```
A>SUBMIT RPED
```

Voraussetzung ist natürlich, daß sich die Datei SUBMIT.COM auf der Arbeitsdiskette befindet. Sonst tippen Sie

```
A>TYPE RPED.SUB
```

und geben die aufgelisteten Befehle von Hand ein.

Drucker perfekt integriert

Eine bisher nahezu unmöglich erscheinende Einbindung eines Druckers in das Betriebssystem realisierten inzwischen die Ingenieure von Amstrad. Nachdem der Drucker zum Lieferumfang gehört, ließen sie einen großen Teil seiner Elektronik einfach weg. Statt dessen steuert der Computer die Druckfunktionen. Während Sie sich bei Verwendung der Locoscript-Textverarbeitung darüber nicht den Kopf zerbrechen müssen, können Sie den Drucker unter CP/M Plus individuell nach Ihren Wünschen konfigurieren. Dazu dient das CP/M-Programm PAPER.COM.

Gestartet wird PAPER.COM durch Eingabe des Schlüsselworts »PAPER«, gefolgt von diversen Parametern. Diese lauten:

- FORM LENGTH n

»Form Length« (Länge des Formulars) legt die Zahl der Zeilen pro Seite fest. Gestattet sind Werte zwischen 6 und 99. Außerdem wird der Zeilenabstand automatisch auf sechs Zeichen pro Zoll eingestellt. Zwischen zwei Seiten erscheinen ab sofort keine Leerzeilen mehr.

- GAP LENGTH n

»Gap Length« (Länge des Abstands) bestimmt, wie viele Zeilen am unteren Rand der Seite freigehalten werden. Der Wert darf zwischen 0 und 99 liegen.

- LINE PITCH n

»Line Pitch« (Punktabstand zwischen den Zeilen) setzt die Zahl der Zeilen pro Zoll fest. Hier sind nur die Werte 6 und 8 gestattet.

- SINGLE SHEET

»Single Sheet« (Einzelblatt) schaltet auf Einzelblattbetrieb um.

- CONTINUOUS STATIONARY

»Continuous Stationary« (Endlospapier) sagt dem Drucker, daß Endlospapier vorliegt.

- PAPER OUT DEFEAT ON
  PAPER OUT DEFEAT OFF

»Paper Out Defeat« (Papierende-Meldung) legt fest, ob der Drucker das Papierende melden soll oder nicht.

Wer in der englischen Sprache nicht so bewandert ist, wird sich oft über die Schreibung einzelner Befehle nicht ganz sicher sein: Heißt es jetzt eigentlich »CONTINUOUS« oder »CONTINOUS«? Kürzen Sie dann doch einfach die Befehle ab! PAPER.COM versteht folgende Kurzbezeichnungen:

- »F« für »FORM LENGTH«
- »G« für »GAP LENGTH«
- »L« für »LINE PITCH«
- »S« für »SINGLE SHEET«
- »C« für »CONTINUOUS STATIONARY«
- »P« für »PAPER OUT DEFEAT«

Um die Druckersteuerung möglichst nicht in Rechnerei ausarten zu lassen, gibt es zwei vorgegebene Standardblätter für die Papierformate DIN A4 und DIN A5:

```
A> PAPER A4
```

Der Computer bedruckt das Papier nun als Einzelblatt des Formats DIN A4. Auf eine Seite passen 70 Zeilen, am unteren Rand werden drei Zeilen freigehalten.

```
A> PAPER A5
```

Der Joyce nimmt nun DIN-A5-Papier an: 50 Zeilen pro Seite und drei Leerzeilen am unteren Blattrand.

Ein zusätzlicher PAPER-Parameter gilt für Endlospapier. Wenn nur eine Zahl zwischen 1 und 17 genannt wird, nimmt der Computer folgende Bedingungen an:

- Es werden Endlosblätter verwendet.
- Die Seitenlänge beträgt »n« Zoll.
- Der Zeilenabstand liegt bei sechs Zeilen pro Zoll.
- Am unteren Blattrand werden keine Zeilen freigehalten.
- Das Papierende wird dem Computer mitgeteilt.

Wir hoffen, Ihnen fällt der Umgang mit dem Joyce durch unseren kleinen Überblick jetzt ein wenig leichter.

(Martin Kotulla/Udo Reetz)

MAC und RMAC – zwei ungleiche Brüder

Wer in Maschinensprache programmieren will, benötigt einen leistungsstarken Assembler und Hilfsprogramme. Beim CP/M 3.0 sind sie »kostenlos« dabei.

CP/M war bereits seit seiner Entstehung hauptsächlich als Betriebssystem für Programmierer gedacht. Die Version 3.0 fiel vor allem erheblich bedienungsfreundlicher aus und zeigte sich nachsichtiger gegenüber Nicht-Fachleuten. Dabei blieben die guten Hilfen zur Programmierung nicht nur erhalten, sondern erfuhren sogar noch weitere Verbesserungen.

In näherem oder weiterem Sinn gehören folgende Programme unter CP/M Plus in die »Abteilung Programmiertools« und werden von Schneider beim CPC 6128 und beim Joyce PCW 8256/8512 mitgeliefert. Als erstes sind wohl MAC.COM, der CP/M Plus- Standardassembler, und RMAC.COM, sein »Bruder«, der relokalisierbaren Code erzeugt, erwähnenswert.

HEXCOM.COM übersetzt die von MAC.COM erzeugten HEX-Files in direkt ausführbare COM-Dateien. Link-80 (auf der Diskette unter dem Dateinamen LINK.COM zu finden) macht etwas ähnliches für RMAC-Dateien, die die Extension ».REL« besitzen.

XREF.COM formt aus der Druckdatei und der Symboltabelle von mit MAC oder RMAC übersetzten Programmen ein Listing mit Querverweisen (»cross references«). Dieses Listing ist eine große Hilfe bei der Programmdokumentation.

Bei SID.COM (»Symbolic Instruction Debugger«) handelt es sich um einen sehr komfortablen Maschinensprache-Monitor und Programm-Debugger. Er arbeitet im Gegensatz zu DDT.COM von CP/M 2.2 (»Dynamic Debugging Tool«) auch mit den vom Assembler erzeugten Symboltabellen. Zu SID.COM gehören die Utility-Programme TRACE.UTL und HIST.UTL, die beim Testen von Software oft recht nützlich sind.

LIB.COM ist ein »Library«-Programm. Wenn Sie viel in Maschinensprache oder anderen Compilersprachen (zum Beispiel PL/I-80, Pascal/MT+ oder CBASIC) unter CP/M programmieren, können Sie wichtige und häufig benötigte Unterprogramme in Objektcode-Form als Programmbibliotheken anlegen. LIB.COM verwaltet diese Libraries.

Die vier verbleibenden Programme braucht man seltener: DUMP.COM listet Diskettendateien auf dem Bildschirm in hexadezimaler Form auf. Darstellbare ASCII-Zeichen werden direkt gezeigt.

GENCOM.COM dient zur Anbindung von RSX-Modulen an COM-Files. »RSX« steht für »Resident System Extension«, genau wie die RSX-Befehle im Schneider-Basic. Beide »RSX«-Systeme haben aber absolut nichts miteinander zu tun. Die RSX unter CP/M ist vielmehr eine Betriebssystem-Erweiterung, die über den BDOS-Einsprung aufgerufen werden. Sie binden sich in die BDOS-Kette ein und merken automatisch, wann sie angesprochen werden sollen. »RSXen« erfordern aber sehr spezielle Programmierkenntnisse, die einiges an Erfahrung mit dem CP/M-Betriebssystem voraussetzen.

SAVE.COM ersetzt den vormals in der CP/M-Version 2.2 residenten Befehl SAVE, der Teile der TPA auf der Diskette abspeichert. SAVE.COM arbeitet mit SID.COM zusammen; Sie lernen dieses Programm später noch ausführlicher kennen.

PATCH.COM vermerkt im Kopf einer Datei eine Nummer zwischen 0 und 31. PATCH dient dazu, Dateien mit Versionsnummern zu bezeichnen.

Betrifft: Assembler

Die beiden Assembler MAC.COM und RMAC.COM haben einen großen Nachteil: Sie verarbeiten nur den mnemonischen Code des 8080/8085-Mikroprozessors. Z80-Programmierer müssen also umdenken. Ansonsten bieten die beiden alles, was man sich nur wünschen kann. Vor allem – daher rührt auch ihr Name – sind sie makrofähig. Makros sind Befehlsgruppen, die unter einem bestimmten Namen zusammengefaßt und später anhand dieses Namens wieder aufgerufen werden können.

MAC und RMAC sind aufwärtskompatibel zum Standardassembler ASM.COM, der zum Lieferumfang von CP/M 2.2 gehört. Das bedeutet, daß sie alle Programme, die ASM übersetzt, auch verarbeiten. Andersherum geht es natürlich nicht. Man kann ASM keine Programme vorsetzen, die spezielle Fähigkeiten von MAC oder RMAC ausnutzen.

Doch alles der Reihe nach. Gehen wir zuerst einmal auf die Grundlagen von MAC und RMAC ein.

Dem Assembler geben Sie eine Diskettendatei vor, die den Quellcode im Klartext enthält. »Klartext« heißt, daß die Datei keine Sonderzeichen oder spezielle Codierungen aufwärts. Wordstar ist im normalen D-Modus dafür also herzlich ungeeignet. Verwenden Sie dieses Textprogramm bei der Erstellung von Maschinencode-Programmen (wie übrigens auch bei allen anderen Programmdateien) grundsätzlich nur im N-Modus. Dann weiß Wordstar nämlich, daß Sie eine Quellcode-Datei erstellen und erzeugt reine ASCII-Files. Wer keinen geeigneten Texteditor besitzt, kann auf die Basic-»Simpel«-Lösung zurückgreifen:

```
10 OPENOUT "PROG.ASM"
20 PRINT #9,"START MVI A,9"
30 PRINT #9,"      MOV C,A"
40 PRINT #9,"      LXI D,STR$"
50 PRINT #9,"      CALL 5"
60 PRINT #9,"      JMP  0"
70 PRINT #9,"STR$  DB 'Hallo!$'"
80 CLOSEOUT
```

Allzu komfortabel ist das natürlich nicht. Es handelt sich eben um eine Notlösung.

Eine typische Quellcode-Zeile besteht aus einem oder mehreren der folgenden Einzelglieder: Zeilennummer, Label, Befehl, Operanden, Kommentar.

Selbstverständlich können Sie die Zeilennummer auch weglassen. Der Assembler ignoriert sie ohnehin. Sie dient nur dazu, Kompatibilität zu möglichst vielen Texteditoren herzustellen.

Einige Beispiele für syntaktisch korrekte Quellcode-Zeilen mit Befehlen und Operanden:

```
MVI    A,3
ADI    23
JMP    0
CALL   5
LHLD   30H
```

Manche 8080-Befehle benötigen überhaupt keine Operanden. In diesem Fall ist das Operandenfeld leer:

```
SPHL
RNZ
XTHL
```

Um in anderen Maschinencode-Befehlen auf eine Speicheradresse Bezug zu nehmen, verwendet man symbolische Adressen, sogenannten »Labels«. Sie können diese jedem beliebigen Maschinencode-Befehl voranstellen.

```
START MVI    A,23
      JMP    ENDE
      MVI    C,3
      CALL   5
ENDE  JMP    0
```

Das Label, auch »Marke« genannt, besitzt dann als Wert die Adresse des zugehörigen Maschinenbefehls. Steht etwa das obige »MVI A,23« an der Adresse 2710 hex, hat »START« den Wert 2710 hex. Mit JMP, CALL oder LHLD sprechen Sie zusammen mit dem Label die Adresse 2710 hex an.

Eine Marke darf bis zu 16 Buchstaben lang sein. Dabei sind alle Zeichen signifikant, werden also bis zur 16. Stelle hin unterschieden.

Namen frei gewählt

Bei der Namensgebung sind Sie relativ ungebunden. Buchstaben, Zahlen, Fragezeichen und der »Klammeraffe« sind erlaubt und werden als Stelle gezählt. Lediglich das ebenfalls zugelassene Dollarsymbol (»$«) überliest der Assembler und eignet sich demnach recht gut zur Gliederung langer Namen. Die Labels »LANGER$NAME« und »LANGERNAME« sowie »LANGERNAM$E« sehen für den Assembler gleich aus.

Wenn Sie wollen, dürfen Sie hinter die Labels einen Doppelpunkt setzen. MAC und RMAC verstehen die Zeilen aber auch ohne:

```
LOAD    LXI    D,3FACH
```

oder

```
LOAD:   LXI    D,3FACH
```

Kommentare hängen Sie am besten hinter der Programmzeile an. Ein Strichpunkt oder ein Stern trennt sie vom übrigen Text ab. Sie dürfen auch allein in einer Zeile stehen:

```
START LXI SP,0100H ; Stackpointer auf TPA
* Hier beginnt das Hauptprogramm
; Und hier sind die Daten
```

Numerische Konstanten in Ausdrücken lassen sich in verschiedenen Zahlensystemen angeben: dezimal, binär, hexadezimal und sogar oktal!

Dezimale Konstanten enden wahlweise mit einem »D«: 32767D entspricht 32767, man kann das D also auch weglassen.

Hexadezimale Werte besitzen als Kennung ein »H«: 3FACH, 2710H, 7FH. Beginnt eine Hexzahl mit einem Buchstaben (»A« bis »F«), sollten Sie ihr eine Null voranstellen, damit der Assembler sie von einem Label unterscheiden kann. Statt »FFFFH« schreiben Sie besser »0FFFFH«.

Binärzahlen enden mit einem »B« für »Binary«. Gerade bei diesen Zahlen-»Bandwürmern« empfiehlt es sich, sie mit Dollarzeichen zu gliedern. »0000111101011010B« ist ein ziemlich abschreckendes Beispiel für eine – zulässige – Binärkonstante, während »0000$1111$0101$1010$B« durch die Unterteilung in Nibbles (Halb-Bytes) durchaus leserlich ist.

Oktale Konstanten werden Sie zwar kaum jemals benötigen, aber der Vollständigkeit halber: Sie enden mit einem »o« oder »O«, dem Buchstaben, keinesfalls der Null! Oktalzahlen, lassen sich aber auch mit dem weniger mißverständlichen »Q« oder »q« kennzeichnen:

```
3377o = 33770 = 3377Q = 3377q
```

Auch mit Zahlen und Symbolen kann man rechnen und in Ausdrücken Formeln erstellen. Neben den vier Grundrechenarten (Addition »+«, Subtraktion »-«, Multiplikation »*« und der Division »/«) stehen auch die sogenannten unären Operatoren zur Verfügung:

```
+X=X, -X=0-X.
```

Während »/« eine Integerdivision durchführt, liefert MOD (»modulo«) den Rest dieser ganzzahligen Division.

Der Divisionsrest läßt sich folgendermaßen beschreiben: Wenn Sie 13 durch 4 teilen, erhalten Sie das ganzzahlige Ergebnis 3. »3*4«, also der Umkehrschritt, ergibt aber nicht 13, sondern 12. Die Differenz zwischen 13 und 12 ist der Rest, hier die Zahl 1.

Auch Shift-Befehle verstehen die beiden Assembler. Analog zu den Turbo-Pascal-Operatoren heißen diese auch hier »SHL« (Shift Left) und »SHR« (Shift Right«). Beide schieben die Bits in einem Byte oder 16-Bit-Wort um eine gegebene Anzahl von Stellen nach links oder rechts und füllen die freigewordenen Stellen mit Null-Bits auf.

Eine Linksverschiebung um eine Position entspricht damit der Multiplikation mit Zwei, ein Ein-Bit-Rechtsschieben der Division durch Zwei.

Deutlicher wird das ganze an zwei praktischen Beispielen:

```
001101B SHL 1 ergibt 011010B
001101B SHR 1 ergibt 000110B
```

HIGH und LOW berechnen das High- beziehungsweise Low-Byte einer 16-Bit-Zahl. So ergibt »HIGH 3FACH« den Wert 3F hex und »LOW 3FACH« die Zahl 0AC hex.

Auch einige Logikbefehle, die Sie vielleicht vom Schneider-Basic her kennen, sind als Operatoren zugelassen. »AND«, »OR« und »XOR« verknüpfen jeweils zwei Operanden Bit für Bit.

```
3 AND 12
255 OR X
16000 XOR 16001
3 OR 4 XOR 23 AND 255
```

»NOT« liefert das logische Gegenteil einer Zahl, denn es dreht alle ihre Bits um. Aus Null-Bits werden Einsen, aus Eins-Bits Nullen. Haben Sie beispielsweise die Binärzahl 0101B, ergibt NOT 0101B: 1010B.

Die Wahrheitswerte TRUE (wahr) und FALSE (falsch) erhalten Sie von einigen Vergleichsoperatoren:

a EQ b (»check if equal«: prüfe, ob a gleich b)
So liefert 2 EQ 2 »TRUE«, 2 EQ 3 hingegen »FALSE«.

a NE b (»check if not equal«: prüfe, ob a ungleich b)
50 NE 49 ergibt »TRUE«, während 50 NE 50 der Wahrheitswert »FALSE« ist.

a LT b (»check if less than«: prüfe, ob a kleiner b)
4 LT 3 ergibt »FALSE«, 3 LT 4 im Gegensatz dazu »TRUE«.

a LE b (»check if less than or equal«: prüfe, ob a kleiner oder gleich b ist)
3 LE 4 ist »TRUE«, 4 LE 4 ebenso. 5 LE 2 meldet »FALSE«.

a GT b (»check if greater than«: prüfe, ob a größer als b ist)
20 GT 21 ergibt den booleschen Wert »FALSE«, 21 GT 19 liefert das Resultat »TRUE«.

a GE b (»check if greater than or equal«: prüfe, ob a größer oder gleich b ist)
30 GE 10 ist »TRUE«, ebenso wie 30 GE 30. Dahingegen zeigt 30 GE 40 den Wahrheitswert »FALSE« an.

Wem die Lust fehlt, die englischen Abkürzungen für die Vergleichsbefehle zu erlernen, kann natürlich auch seine Mathematikkenntnisse anwenden und auf die bekannte »Größer-Kleiner«-Symbolik zurückgreifen:

LT	<	Kleiner
LE	<=	Kleiner oder gleich
EQ	=	Gleich
NE	<>	Ungleich
GE	>=	Größer oder gleich
GT	>	Größer

Strings gibt's auch

Stringkonstanten werden in Apostrophe eingeschlossen. Dabei darf ein String pro Zeile maximal 64 Buchstaben enthalten. Ein paar Beispiele für gültige Strings:

```
'A'
'CP/M lernen mit Happy-Computer!'
''
```

Wollen Sie einen Apostroph im String unterbringen, setzen Sie an seiner Stelle das Anführungszeichen ein:

»The lazy dog"s back« ergibt im Objektcode »The lazy dog's back«.

Bestehen Sie partout auf ein Anführungszeichen im Text, verursacht das einige Schwierigkeiten. Geben Sie in diesem Fall den ASCII-Code des Zeichens an, nämlich 34:



```
34,'Bitte sehr',34,' sagte er'
```

»Direktiven«, dieses Wort klingt so nach »Befehl von oben«. Bei den CP/M-Assemblern hat es eine erheblich positivere Bedeutung. »Direktiven« sind wohl den meisten eher geläufig als »Pseudo-Befehle«. Das sind Kommandos, die in den Quellcode wie normale Maschinenbefehle eingefügt werden und beim Assembler verschiedene Reaktionen auslösen können. Ein Beispiel dafür ist:

```
PAGE 53
MVI  A,23
```

Während der MVI-Befehl ein ganz normales Mnemonic ist, handelt es sich bei »PAGE« um die besagte Direktive. Sie teilt dem Assembler schlicht und einfach mit, daß alle Seiten bei der Ausgabe des Druckerlistings 53 Zeilen umfassen sollen.

Direktiven – nicht von oben

Es gibt natürlich auch bedeutendere und häufiger benötigte Direktiven. So sagt »ORG Adresse« dem Assembler, daß er den internen Programmzähler (»Origin«) bei der Übersetzung auf einen bestimmten Wert setzen soll. An die genannte Adresse wird das Programm dann später bei der Ausführung geladen. Für CP/M-Systeme setzen Sie am geeignetsten ausschließlich »ORG 0100H« ein, denn 0100 hex ist der Beginn des CP/M-Programmspeichers der TPA. Diese Feststellung gilt natürlich nur für MAC, nicht für RMAC. Denn RMAC erzeugt verschiebbaren Code, und da ist ein fester ORG-Adreßwert ja eher hinderlich.

Abschließen sollten Sie jedes Programm mit der »END«-Direktive. Alle dem »END« folgenden Befehlszeilen überliest der Assembler, das Dateiende erkennt er aber auch ohne »END«. Denken Sie aber unbedingt daran, bei der Eingabe in den Editor nach der letzten Zeile mindestens einmal ENTER oder RETURN zu drücken. Sollten Sie das vergessen, kann es sein, daß der Assembler ohne Vorwarnung »aussteigt«.

Sie haben bereits die »Labels« kennengelernt. Mit »EQU« können Sie einem Label auch gleich einen festen Wert zuweisen. So liegt beispielsweise die Warmstart-Adresse von CP/M bei 0000 hex. Wollen Sie diese etwa »WARMBOOT« nennen, schreiben Sie diese Zeile:

```
WARMBOOT EQU 0000H
```

Um einen Warmstart des Computers im Programm auszulösen, geben Sie wahlweise »JMP 0000H« oder »JMP WARMBOOT« an. So wird es einsichtig, daß sich die Verwendung von Labels gut eignet, verständliche Programme zu schreiben, und damit sehr zu empfehlen ist. Statt Konstanten sind auch Formeln und Berechnungen rechts vom »EQU« gestattet. Sogar den Bezug auf vorher definierte Labels unterstützen MAC und RMAC:

```
CONSOLE$PORT EQU OFFDBH
JOYST$PORT   EQU CONSOLE$PORT+5*3
```

Ein einmal mit »EQU« definiertes Label läßt sich nicht so ohne weiteres später im Programm ändern. Legen Sie darauf trotzdem Wert, müssen Sie auf »SET« zurückgreifen:

```
CONSOLE   SET  7B00H
          IN   CONSOLE
          PUSH PSW
CONSOLE   SET  630FH
          OUT  CONSOLE
          POP  AF
```

Zuerst besitzt »CONSOLE« den Wert 7B00 hex, später aber 630F hex. Bei »IN« wird die erste Adresse ausgewertet (7B00 hex), bei dem auf das zweite »SET« folgenden »OUT« dagegen 630F hex.

Um Daten direkt in den Quelltext einzusetzen, bedient man sich der Direktive »DB«. Das steht für »Define Byte«. Der Beginn einer ASCII-Tabelle sieht mit »DB« so aus:

```
DB 32,33,34,35,36,37,38,39,40
```

ASCII-Codes sind in Apostrophe einzuschließen:

```
DB ' ','!',34,'#','$','%'
```

Bis zu 64 Buchstaben und Symbole können Sie auch auf einmal angeben:

```
DB ' !#$%&"()*+,-./01234'
```

Der Assembler berechnet die einzelnen Werte oder Formeln und legt sie Byte für Byte im Speicher ab. Bei Adreßausdrücken ist die 8-Bit-Breite von »DB« hinderlich. »DW« (»Define Word«) ist da eher angebracht:

```
DW 3FACH,16300,'AB',0101101B
```

Wollen Sie für einen Datenbereich Speicherplatz reservieren, ohne ihn gleich mit bestimmten Werten vorzubelegen, empfiehlt sich der Pseudobefehl »DS« (»Define Space« oder »Define Storage«). Um etwa 30 Byte für den Prozessorstapel bereitzuhalten, wählen Sie folgende Formulierung:

```
        DS  30
STACK   EQU $
        ORG 0100H
        LXI SP,STACK
```

MAC und RMAC erstellen auf Wunsch bei der Assemblierung eine Listdatei, die den Quellcode und die erzeugten Bytes in übersichtlicher Form darstellt. Mit zwei Assembler-Direktiven läßt sich die Ausgabe steuern. »PAGE« legt fest, mit wievielen Zeilen jede Druckseite beschrieben wird, bis der Assembler ein Form-Feed (Seitenvorschub) ausgibt. »PAGE 40« fügt zum Beispiel nach jeweils 40 Zeilen einen Seitenvorschub ein.

Eine besondere Bedeutung kommt noch dem Befehl »PAGE 0« zu. Er unterdrückt jeglichen Seitenvorschub und weist den Assembler an, alle Listingzeilen ohne Unterbrechung zu drucken. »PAGE« ohne Zahlenparameter löst einen unmittelbaren Seitenvorschub aus.

»TITLE« bestimmt die Meldung, die am Anfang jeder Listingseite die erste Zeile einnimmt. Nach »TITLE« geben Sie eine Stringkonstante an, die von Apostrophen umgeben sein sollte:

```
TITLE 'Diskmonitor V2.0'
TITLE 'Mallard-Basic V4 Rev.1'
```

Der Assembler MAC druckt diesen String zusammen mit seiner eigenen Startmeldung, der Versionsnummer des Assemblers und der Seitennummer aus:

```
CP/M MACRO ASSEM 2.0
#001 Diskmonitor V2.0
CP/M MACRO ASSEM 2.0
#001 Mallard Basic V4 Rev.1
```

Etwas anders sieht das bei RMAC aus:

```
CP/M RMAC ASSEM 1.1
#001 Diskmonitor V2.0
CP/M RMAC ASSEM 1.1
#001 Mallard Basic V4 Rev.1
```

Wer größere Programmpakete unter CP/M zu entwickeln beabsichtigt, schreibt oft verschiedene Versionen, die jeweils an einen speziellen Computer angepaßt sind. Um nun nicht für jede Version eine eigene Quellcode-Datei anlegen zu müssen, kann man den Assembler anweisen, Programmteile nur dann zu übersetzen, wenn eine bestimmte Bedingung zutrifft. Dazu schachteln Sie die bedingt zu übersetzenden Segmente mit »IF« und »ENDIF«.

Stellen Sie sich etwa vor, Sie wollen ein Programm CLS.COM schreiben, das beim Schneider CPC und beim Joyce den Bildschirm löscht und den Cursor in die linke obere Bildschirmecke setzt. Unter CP/M 2.2 verwenden Sie dazu ASCII 12, bei CP/M Plus hingegen ESC-E/ESC-H. Eine Version für CP/M 2.2 sieht so aus:

```
BDOS      EQU  0005H
WARM      EQU  0000H
TPA       EQU  0100H
OUTPUT$   EQU  0009H
          ORG  0100H
          LXI  SP,TPA
          MVI  C,OUTPUT$
          LXI  D,CLS
          CALL BDOS
          JMP  WARM
CLS       DB   12,'$'
          END
```

Unter CP/M Plus heißt es dagegen:

```
BDOS      EQU  0005H
WARM      EQU  0000H
TPA       EQU  0100H
```

```
OUTPUT$  EQU  0009H
         ORG  0100H
         LXI  SP,TPA
         MVI  C,OUTPUT$
         LXI  D,CLS
         CALL BDOS
         JMP  WARM
CLS      DB   27,'E',27,'H$'
         END
```

Und so sieht die Kombination der beiden Listings mit Hilfe der bedingten Assemblierungsmöglichkeit aus:

```
TRUE     EQU   OFFFFH    ; wahr
FALSE    EQU   NOT TRUE  ; unwahr
CPM3     EQU   TRUE      ; Oder
                     CPM3  EQU FALSE
BDOS     EQU   0005H
WARM     EQU   0000H
TPA      EQU   0100H
OUTPUT$  EQU   0009H
         ORG   0100H
         LXI   SP,TPA
         MVI   C,OUTPUT$
         LXI   D,CLS
         CALL  BDOS
         JMP   WARM
         IF    CPM3
CLS      DB    27,'E',27,'H$'
         ENDIF
         IF    NOT CPM3
CLS      DB    12,'$'
         ENDIF
         END
```

Nun erzeugen Sie ganz einfach eine CP/M 3.0- und eine CP/M 2.2-Version. Sie brauchen nur

```
CPM3     EQU   TRUE
```

für die CP/M 3.0-Version oder

```
CPM3     EQU   FALSE
```

für die CP/M 2.2-Version einzusetzen.

Etwas übersichtlicher gestaltet die ganze Sache die Konstruktion »IF – ELSE – ENDIF«. Das Clearscreen-Programm läßt sich mit ELSE so darstellen:

```
TRUE     EQU   OFFFFH
FALSE    EQU   NOT TRUE
CPM3     EQU   TRUE    ; Oder
                     CPM3 EQU FALSE
BDOS     EQU   0005H
WARM     EQU   0000H
TPA      EQU   0100H
OUTPUT$  EQU   0009H
         ORG   0100H
         LXI   SP,TPA
         MVI   C,OUTPUT$
         LXI   D,CLS
         CALL  BDOS
         JMP   WARM
         IF    CPM3
CLS      DB    27,'E',27,'H$'
         ELSE
CLS      DB    12,'$'
         ENDIF
         END
```

Doch seien Sie sicher; allzu häufig werden Sie auf derartige Finessen nicht zurückgreifen.

Bis jetzt wissen Sie noch gar nicht, wie Sie MAC oder RMAC dazu bringen, ein Programm in Maschinencode zu übersetzen. Gehen wir davon aus, Sie haben eines der CLS-Listings abgetippt und als CLS.ASM gespeichert. Die Extension ».ASM« im Dateinamen ist von äußerster Wichtigkeit, denn sonst findet der Assembler die Datei nicht.

Sie können es sich nun einfach machen und MAC nur mit dem Dateinamen laden:

```
A>MAC CLS
```

Nach kurzer Wartezeit erscheinen einige Meldungen auf dem Bildschirm:

```
CP/M MACRO ASSEM 2.0
0110
000H USE FACTOR
END OF ASSEMBLY
```

Schauen wir uns die Bedeutung der einzelnen Textzeilen an: »CP/M MACRO ASSEM 2.0« ist schlicht die Startmeldung, die darüber Auskunft gibt, daß MAC jetzt aktiv ist. Die Zeile »0110« (eventuell bei Ihnen auch »0113«) zeigt die letzte vom Programm belegte Adresse. Das Maschinencode-Programm wird im Speicher also zwischen 0100 (dem normalen Beginn der TPA) und 0110 beziehungsweise 0113 hex (der Angabe von MAC) liegen.

HEXCOM stellt sich vor

Der »Use Factor« ist eine Information des Assemblers, die die Auslastung des Arbeitsspeichers mit der Symboltabelle während der Assemblierung im Verhältnis zur Größe des Gesamtspeichers beschreibt. Liegt der Auslastungsfaktor nahe 0FF hex, bedeutet dies, daß der Assembler bald nicht mehr in der Lage sein wird, das Programm zu übersetzen. Bei solchen Miniprogrammen wie CLS ist der Speicher natürlich praktisch kaum belastet. Deshalb ergibt sich für den Belegungsfaktor der angegebene Wert 0.

Wenn Sie jetzt das Directory listen, entdecken Sie einige neue Dateien. Da wäre zuerst einmal CLS.PRN, die Listingdatei. Sie läßt sich so auf dem Bildschirm ausgeben:

```
A>TYPE CLS.PRN
```

CLS.SYM ist die Symboltabelle. In diese Datei schreibt MAC alle Symbole zusammen mit ihrem Wert:

```
A>TYPE CLS.SYM
     0005 BDOS
     010E CLS
     0009 OUTPUT
     0100 TPA
     0000 WARM
```

Neben der besseren Programmdokumentation erfüllt sie noch eine weitere Aufgabe. Der Debugger SID kann diese Tabelle nämlich auch lesen, wodurch Sie in diesem Monitorprogramm mit symbolischen Adressen arbeiten können. Doch dazu später mehr.

Nach einem »MAC-Lauf« enthält die Diskette noch kein startfähiges Programm, nur eine sogenannte »Intel-Hex-Datei«. Das sind kryptische Kombinationen von Hexadezimalzahlen. CLS.HEX könnte bei Ihnen vielleicht so aussehen:

```
:10010000310001OE09110E01CD0500C
300000C24C1 :0000000000
```

Dem Menschen erscheinen diese Dateien unverständlich, während HEXCOM.COM damit ganz ausgezeichnet zurechtkommt. Es erzeugt aus HEX-Dateien COM-Programmdateien. Sie rufen das Konvertierungsprogramm zusammen mit dem Dateinamen des zu übersetzenden Programmes, aber ohne Extension auf:

```
A>HEXCOM CLS
```

Es stellt sich Ihnen kurz vor:

```
HEXCOM VERS: 3.00
```

und verrichtet dann seine Arbeit. Danach gibt es Informationen über CLS.COM aus:

```
FIRST ADDRESS 0100
LAST  ADDRESS 010F
BYTES READ    0010
RECORDS WRITTEN 01
```

Der Programmstart liegt also bei 0100 das Programmende bei 010F hex. HEXCOM hat 0010 hex (16) Byte gelesen und einen 128-Byte-Record auf die Diskette geschrieben.

Sie können das Programm CLS.COM nun durch Eingabe des Dateinamens starten:

```
A>CLS
```

Nun nehmen wir uns das Programm RMAC vor. Besonders viel Sinn gibt das hier noch nicht, weil wir nicht mehrere Programm-Module zusammen-»linken«. Gerade das ist aber die Spezialität von RMAC. Starten läßt sich RMAC auf dieselbe Art und Weise, die Sie schon von MAC kennen:

```
A>RMAC CLS
CP/M RMAC ASSEM 1.1
0110
000H USE FACTOR
END OF ASSEMBLY
```

RMAC legt – ebenso wie MAC – eine SYM- und eine PRN-Datei an. Nur das ».HEX« fehlt. Dafür gibt es ».REL«, ein sogenanntes »Relocatible File«. Das REL-Format ist Microsoft-kompatibel. Das heißt, daß sich RMAC-Programme mit Programmen verschiedener Microsoft- und Digital-Research-Compiler verketten lassen. Link-80 erledigt das für Sie. Geben Sie nur eine einzige Link-Datei an, verhält sich LINK fast wie HEXCOM:

```
A>LINK CLS
LINK 1.31
ABSOLUTE     0000
CODE SIZE    0110 (0100-020F)
DATA SIZE    0000
```

```
COMMON SIZE   0000
USE FACTOR    00
```

Auch von RMAC und LINK produzierte Programme werden wie üblich gestartet:

```
A>CLS
```

Nun wäre es ja richtig »langweilig«, wenn alle Assemblerprogramme immer auf genau dieselbe Art übersetzt würden. MAC und RMAC besitzen deshalb eine Reihe von Optionen, die die Übersetzung steuern. Diese stehen jeweils hinter dem Dateinamen und sind von ihm durch ein Dollarsymbol getrennt. Sie bestehen stets aus zwei Buchstaben. Eine typische Aufrufzeile sieht so aus:

```
A>MAC DATEI $AB PA SB
```

»A« bestimmt das Laufwerk mit der ASM-Datei. »AB« läßt den Assembler die Suche also auf dem Laufwerk B beginnen. »AA« sucht auf A. »PA« schreibt die Listdatei auf das Laufwerk A, »SB« die Symboltabelle nach B. Einige wichtige Zusätze:

Ad Diskette (d) für die gelesene ASM-Datei

Hd Diskette (d) für die erzeugte HEX-Datei

Pd Diskette (d) für die Listdatei (».PRN«)

Sd Diskette (d) für die Symboltabelle (».SYM«)

Neben den Standardlaufwerken A bis O (beim Schneider nur A und B, allenfalls bei RAM-Disks oder Festplatten gibt es weitere Laufwerksbezeichnungen) versteht der Assembler auch einige »Scheingeräte«. So fühlt sich bei »P« der Drucker (Printer) und bei »X« der Bildschirm angesprochen. Führen Sie »Z« (Zero Device) auf, unterdrückt dies die Ausgabe. Um also keine Hexdatei zu erzeugen, die Listdatei auf den Bildschirm zu schreiben und die Symboltabelle auf dem Drucker auszugeben, geben Sie ein:

```
A>MAC DATEI $PX HZ SP
```

RMAC versteht diese Zusätze ebenfalls, allerdings gilt statt der Bezeichnung »H« für Hexfiles hier »R« für die RMAC-typischen REL-Dateien.

Eine komplette Tabelle aller MAC- und RMAC-Parameter finden Sie in der Tabelle.

Ein »richtiger« Assembler muß Makros verarbeiten – so zumindest die Meinung vieler Anwender. So ausnehmend wichtig sind sie jedoch bestimmt nicht, aber oft ganz nützlich.

Was sie tun und wie man sie anwendet, verdeutlicht am ehesten ein Beispiel. Will man in CP/M einen String auf dem Bildschirm ausgeben, benutzt man die BDOS-Funktion 9:

```
          LXI  D,STRING
          PUSH PSW
          PUSH B
          PUSH D
          PUSH H
          MVI  C,9
          CALL BDOS
          POP  H
          POP  D
          POP  B
          POP  PSW
```

Wenn Sie des öfteren Strings auf dem Bildschirm ausgeben müssen, programmieren Sie eine Unter-Routine:

```
HAUPT     LXI  D,STRING
          CALL AUSGABE
          LXI  D,STRING2
          CALL AUSGABE
          JMP  $

AUSGABE   PUSH PSW
          PUSH B
          PUSH D
          PUSH H
          MVI  C,9
          CALL BDOS
          POP  H
          POP  D
          POP  B
          POP  PSW
          RET
```

Während das natürlich – von der Programmlänge her betrachtet – eine optimale Lösung darstellt, schaut es beim Geschwindigkeitsverhalten der Routine schlechter aus. Denn zum normalen Programmcode kommt noch der Unterprogramm-Aufruf und die Rückkehr zum Hauptprogramm. Gerade Befehle, die den Z80/8080-Stack benötigen und sich zusätzlich noch eine 16-Bit-Adresse aus dem Speicher holen müssen, sind besonders langsam. So dauert die Bearbeitung eines Befehls wie »LD A,B« (MOV A,B) beim Z80 vier Taktzyklen, ein Unterprogramm-Aufruf aber 17. Und der RET-Befehl schlägt mit zehn weiteren Taktzyklen zu Buche. Insgesamt also 27 »unnötige« Zyklen.

Nun könnte man sich mit Blockkopier-Operationen des Texteditors die entsprechenden Programmteile hin- und herkopieren, so daß am Schluß bei jeder Gelegenheit, bei der ein String gedruckt werden soll, der komplette Programmcode dasteht. Unter dem Gesichtspunkt der Übersichtlichkeit ist das aber auf gar keinen Fall zu empfehlen. Diese Lösung, die auf die Makros zurückgreift, ist da schon wesentlich besser.

```
AUSGABE MACRO
        PUSH PSW
        PUSH B
        PUSH D
        PUSH H
        MVI  C,9
        CALL BDOS
        POP  H
        POP  D
        POP  B
        POP  PSW
        ENDM

HAUPT   LXI  D,STRING
        AUSGABE
        LXI  D,STRING2
        AUSGABE
        JMP  $
```

Der Assembler setzt bei der Übersetzung in Maschinencode an allen Stellen, an denen »AUSGABE« steht, den kompletten Programmcode ein, den Sie zwischen »AUSGABE MACRO« und »ENDM« finden.

Das ist zwar sehr speicherplatzaufwendig, aber die zeitgünstigste Lösung. Ganz allgemein läßt sich sagen, daß Makros zu längeren, aber schnelleren Programmen führen, während Unterprogramm-Aufrufe kürzere und langsamere Programme bewirken. Es gilt, stets einen Kompromiß zwischen beiden Extremen zu finden.

Geben Sie bei Makrodefinitionen Kommentare an, die mit einem Strichpunkt eingeleitet werden, setzt der Assembler die Kommentare bei allen Makroaufrufen mit ein. Das belastet natürlich den Speicher sehr:

```
AUSGABE MACRO     ; Zeichenausgabe
        MVI  C,2
        CALL BDOS
        ENDM
```

Sollen Kommentare nur bei der Makrodefinition gelistet werden, nicht aber bei allen aufgelösten Makros, beginnen Sie mit zwei Strichpunkten:

```
AUSGABE MACRO    ;; Zeichenausgabe
        MVI  C,2
        CALL BDOS
        ENDM
```

Wenn Sie innerhalb einer Makrodefinition Sprungmarken verwenden, ist es sinnvoll, diese als lokal zu erklären, um Überschneidungen zu vermeiden. Listen Sie diese Labels im Kopf des Makros zusammen mit dem Schlüsselwort »LOCAL«.

```
AUSGABE MACRO
        LOCAL AUFRUF
        MVI  C,2
AUFRUF  CALL BDOS
        ENDM
```

Damit Sie wirklich etwas von den Makros haben, finden Sie hier eine kleine Sammlung Z80-ähnlicher Befehle:

```
LDIR      MACRO
          DB    OEDH,OBOH
          ENDM
LDDR      MACRO
          DB    OEDH,OB8H
          ENDM
LDI       MACRO
          DB    OEDH,OAOH
          ENDM
LDD       MACRO
          DB    OEDH,OA8H
          ENDM
NEG       MACRO
          DB    OEDH,O44H
          ENDM
```

A	Diskette für das ASM-File
H	Diskette für die HEX-Datei (bei RMAC statt dessen »R«)
L	Diskette für die LIB-Dateien
M	Ausgabe der Makros in die PRN-Datei
P	Diskette für die PRN-Datei
Q	Ausgabe der lokalen Symbole
S	Diskette für die SYM-Datei
1	Listing von Pass 1
+L	Zeilen aus der Makrobibliothek auflisten
-L	Makrozeilen nicht auflisten
+S	Symboltabelle an das Ende der PRN-Datei schreiben
-S	Ausgabe der Symboltabelle unterdrücken
+M	Alle Makrozeilen auflisten
-M	Auflistung von Makrozeilen unterbinden
+Q	Auflistung der lokalen Symbole in Makros
-Q	Unterdrückung der Auflistung
+1	Auflistung der Datei während des First-pass
-1	Unterdrückung der Ausgabe

Tabelle. Alle Mac- und RMac-Parameter auf einen Blick

```
RETN    MACRO
        DB      0EDH,045H
        ENDM
RETI    MACRO
        DB      0EDH,04DH
        ENDM
RRD     MACRO
        DB      0EDH,067H
        ENDM
RLD     MACRO
        DB      0EDH,06FH
        ENDM
```

Sie können diese Makros nun wie die bekannten Z80-Befehle verwenden.

```
        LDIR
        LDDR
        LDI
        LDD
        NEG
        RETN
        RETI
        RRD
        RLD
```

Um beispielsweise einen Speicherblock in 8080-Assembler zu kopieren, müssen Sie folgendes programmieren.

```
        LXI     D,4000H
        LXI     H,8000H
        LXI     B,0300H
        LDIR
```

Das ist natürlich etwas ungewöhnlich, aber es klappt.

Sie können sich auch eine große Programmbibliothek aus Makros aufbauen, zum Beispiel mit einer Reihe verschiedener BDOS- und BIOS-Funktionen. Das Schöne an den Makros ist ja, daß sie nur dann in den Programmcode eingebunden werden, wenn sie wirklich benötigt werden. So müssen Sie aus Ihrer Makrobibliothek nicht mühsam die entsprechenden Teile extrahieren. Makrobibliotheken sollten Sie mit der Extension ».LIB« versehen, zum Beispiel »Z80.LIB«, »DISKUTIL.LIB« oder »MCODE.LIB«. Eingebunden werden sie in den Quellcode über den Befehl »MACLIB«:

```
        MACLIB Z80
        MACLIB DISKUTIL
        MACLIB MCODE
```

Makros ohne Parameter aufzurufen, ist eher umständlich. Denn diese müssen vorher in Register oder Speicherstellen geladen werden – nicht gerade ein Muster an Flexibilität.

Doch die Makros von MAC und RMAC sind so leistungsfähig, daß sie auch Parameter verarbeiten. Sie geben hinter dem Schlüsselwort »MACRO« sogenannte formale Parameter an, deren Zahl mit der Anzahl der Parameter im Makroaufruf übereinstimmen sollte:

```
ZEICHEN     MACRO     N
            MVI       C,2
            MVI       E,N
            CALL      BDOS
            ENDM

HAUPT       ZEICHEN   72
            ZEICHEN   65
            ZEICHEN   80
            ZEICHEN   80
            ZEICHEN   89
            JMP       0
```

Es handelt sich bei Makros um einfaches Einsetzen von Zeichenketten. Also können Sie statt der Zahlen auch Einzelbuchstaben in Apostrophen eingeben:

```
ZEICHEN     MACRO     N
            MVI       C,2
            MVI       E,N
            CALL      BDOS
            ENDM

HAUPT       ZEICHEN   'H'
            ZEICHEN   'A'
            ZEICHEN   'P'
            ZEICHEN   'P'
            ZEICHEN   'Y'
            JMP       0
```

Auf diese Art können Sie dem Makro eine größere Anzahl von Parametern übermitteln. Trennen Sie diese durch Kommata ab:

```
ZPRINT 'A','B',23,'TEXT',CR,LF+3
```

RMAC besitzt einige zusätzliche Direktiven, die zur Unterstützung der modularen Programmentwicklung dienen. ASEG weist den Assembler an, absoluten Programmcode zu erzeugen, dessen Adresse ORG festlegt:

<label> ASEG

CSEG sagt RMAC.COM, den »Code Location Counter« zu verwenden. Die Speicheradressen werden dann erst vom Programmbinder LINK-80 (LINK.COM) bestimmt:

<label> CSEG

DSEG teilt dem Assembler mit, den »Data Location Counter« zu benutzen. Die absoluten Speicheradressen legt dann der Linker fest:

<label> DSEG

PUBLIC erklärt ein Label als »allgemein«. Ein solches Label erscheint im REL-Code und ist von anderen Programm-Modulen über EXTRN zu bestimmen:

PUBLIC label

Zum Beispiel: PUBLIC UPCASE, LOCASE

EXTRN definiert eine Sprungmarke in der Art, daß der Linker sie aus einem anderen Programm-Modul heraussucht. Dies geschieht nur mit PUBLIC-Labels. So steht beispielsweise in CODE1.ASM:

```
START       MVI     A,3
            RET
            PUBLIC START
            END
```

Und in CODE2.ASM:

```
            EXTRN   START
            JMP     START
```

Die beiden Dateien werden von RMAC getrennt assembliert:

```
A>RMAC CODE1
A>RMAC CODE2
```

Erst der Linker verbindet sie:

```
A>LINK CODE1,CODE2
```

Der Aufruf erfolgt dann als CODE1.COM:

```
A>CODE1
```

Bei der Programmdokumentation besonders wertvoll ist ein detailliertes und übersichtliches Listing. XREF.COM kann da helfen. Es produziert aus den von MAC und RMAC erzeugten Dateien ».PRN« und ».SYM« eine neue mit der Extension ».XRF«:

```
A>MAC CLS
A>XREF CLS
```

In der XRF-Datei finden Sie ein Listing des Programms, versehen mit Zeilennummern. Am Ende stehen dann alle verwendeten Labels und Symbole mit ihrem Wert, der Zeile, in der sie definiert sind, und allen Zeilen, in denen sie aufgerufen werden:

```
BDOS        0005   1#     10
CLS         010E   9      13#
OUTPUT      0009   4#     8
TPA         0100   3#     7
WARM        0000   2#     11
```

Das Doppelkreuz bezeichnet die Zeile der Definition.

Um die Ausgabe von der Diskette auf den Drucker umzulenken, können Sie den $P-Schalter angeben:

```
A>XREF CLS $P
```

Hier schließen wir unsere Beschreibung der Assembler und Linker ab.

(Martin Kotulla/ja)



SID – der freundliche Helfer

Wenn Sie bereits unter CP/M 2.2 programmiert haben, wird Ihnen der Systemdebugger DDT bekannt sein. Er erfuhr in der CP/M-Version 3.0 oder auch CP/M Plus weitere Verbesserungen und heißt nun SID.COM.

SID steht für »Symbolic Instruction Debugger«. Das stellt auch schon seine bedeutendste Eigenschaft heraus: Er kann mit Symbolen arbeiten. Doch zuerst zu den »normalen« Fähigkeiten. Gestartet wird SID durch Eingabe der drei Buchstaben »SID«, wahlweise mit einem angehängten Dateinamen.

Die Befehle setzen sich stets aus einem oder zwei Buchstaben zusammen. Ihnen können verschiedene Zahlenwerte folgen. Zu den normalen Zahlen – also solche ohne spezielles Kennzeichen – zählen bei SID.COM hexadezimale Werte, zum Beispiel 40, 4F, 3FAC oder FF3E hex. Im Gegensatz zu MAC und RMAC verlangt SID bei nicht vierstelligen Zahlen keine führenden Nullen. Dezimale Zahlen sind ausdrücklich mit einem Doppelkreuz zu kennzeichnen, etwa #30, #40 oder #255. Sogar Zeichensymbole und Buchstaben können Sie direkt angeben, sofern Sie diese mit Apostrophen umgeben: 'A', 'BC', '*=', '"'. Auch Anführungszeichen in Zeichenketten bleiben somit erhalten ('"').

Die Befehle des SID sind nicht kompliziert, aber sehr vielseitig.

»A« startet den eingebauten Zeilenassembler. Dieser versteht alle 8080-Mnemonics. Direktiven oder Makros verarbeitet er nicht. Für kleine Codierungsaufgaben und zum Abändern kurzer Programmteile eignet er sich aber ohne weiteres.

Um beispielsweise ein Programm an der Adresse 0100 hex einzugeben, tippen Sie:

```
#A0100
0100  MVI  C,9
0102  LXI  D,0300
0105  CALL 5
0108  RST  6
0109  .
```

Durch Eingabe eines Punktes – oder einer Leerzeile – gelangen Sie ins Hauptprogramm zurück. Den Restart 6 belegt SID in der Weise, daß Sie im Hauptprogramm weiter arbeiten können. In diesem kleinen Programm hätte RET dieselbe Bedeutung. Ist das aufgerufene Programm aber ein Unterprogramm, führt Sie RET nur an dessen übergeordnete Aufrufstelle, nicht aber in SID zurück.

Eine Form des Assembleraufrufs erfordert keine Adreßangabe. Der Assembler benutzt dann die zuletzt bei Trace, Auflistung von Speicherauszügen oder Assemblereingaben verwendete Adresse. Haben Sie danach noch keine weiteren Befehle eingegeben, fährt der Assembler in unserem Beispiel bei 0109 hex fort:

```
#A
0109 JMP 0
010C .
```

Sollte einmal der Speicherplatz knapp werden, können Sie den Assembler und Disassembler aus dem Monitor »herauswerfen«. Dazu stellen Sie dem »A« ein Minuszeichen voran:

```
#-A
#A0100
?
```

»C« ruft ein Unterprogramm an einer angegebene Adresse auf.

Wollen Sie das BC-Register mit einem bestimmten Wert versehen, hängen Sie ihn – durch ein Komma abgetrennt – an den C-Befehl an:

```
#C0100,3FAC
```

Und auch das DE-Register kann mit einem 16-Bit-Wert geladen werden:

```
#C0100,3FAC,DEDE
```

»D« steht für »Dump« oder auch »Display Memory«. Der D-Befehl listet Speicherinhalte hexadezimal und als ASCII-Zeichen auf. Sie können ihn ohne Parameter verwenden; dann gibt er die auf die zuletzt verwendete Adresse folgenden 192 Byte aus. Entweder mit Start oder mit Start- und Endadresse grenzen Sie Speicherbereiche, die aufzulisten sind, ein:

```
#D0300
#D#4096
#DFF23,FFFF
#D'AB',34+#0200
```

Sogar einfache Berechnungen, wie Addition und Subtraktion, sind bei Adressen mit SID.COM zulässig.

Gefällt Ihnen die »Zerstückelung« in einzelne Bytes nicht, können Sie SID.COM anweisen, 16-Bit-Adreßworte auszugeben. Dazu hängen Sie an das »D« für »Display Memory« ein »W« für »Words« an. Alle Optionen bleiben erhalten:

```
#DW
#DW0300
#DW0400,'AA'+#24
```

Im Wortformat vertauscht SID 8080- und Z80-gemäß das Low- und Highbyte. Steht im »Byteformat» der Code »03 CA« im Speicher, ändert er sich bei Words in »CA03«.

»F« füllt einen Speicherbereich mit einem speziellen Wert. Dieses Kommando verlangt stets drei Parameter: die Start- und die Endadresse und das einzusetzende Byte. Um in den Speicher von 1000 bis 3000 hex den Wert 255 schreiben, geben Sie diesen Auftrag an SID:

```
#F1000,3000,#255
```

Aber Vorsicht bei der Anwendung des F-Befehls, denn allzuleicht ist das zu untersuchende Programm oder SID.COM selbst überschrieben! Dann müssen Sie CP/M neu starten, und die ganze Arbeit beginnt wieder von vorne.

»G« (Go) ruft eine Maschinencode-Routine an der gegebenen Adresse auf. Zum Beispiel können Sie das vorhin von Ihnen eingetippte Maschinencode-Programm auch mit »G« aufrufen:

```
#G0100
```

Der Monitor meldet sich mit der letzten ausgeführten Adresse zurück:

```
*0108
```

Sie können beim G-Befehl einen oder zwei Breakpoints angeben. Dort stoppt die Programmausführung und der Computer gibt die Kontrolle an den Monitor zurück.

Wollen Sie zum Beispiel nur die Befehle an den Adressen 0100 bis 0105 hex ausführen. Tippen Sie ein:

```
#G0100,0105
```

Achten Sie aber darauf, daß an der Stoppadresse ein ausführbarer Maschinencode-Befehl steht. An der Adresse 0105 hex finden Sie CD, den Code für CALL nnnn. Geben Sie hingegen 0106 hex als Breakpoint an, steigt der Computer aus oder stoppt erst viel später, weil an 0106 hex der Befehl ja nicht beginnt.

Mit »H« (Hexadecimal Arithmetics) können Sie einfache Berechnungen durchführen lassen. Sie legen einen oder zwei Operanden fest, und der Computer addiert und subtrahiert beide:

```
#H9,6
```

```
000F 0003
```

Da aber statt der Operanden auch Formeln zugelassen sind, ist die folgende Darstellungsform besser zu gebrauchen:

```
#H9+6-3241+34-96
CD6C #52588
```

Auch zur Umrechnung zwischen den Zahlensystemen kann man »H« aufrufen:

```
#H34
0034 #52 '4'
#H'A'
0041 #65 'A'
#H#88
0058 #88 'X'
```

Mit der »I«-Anweisung (Simulate Input Line) können Sie dem getesteten CP/M-Programm eine Befehlszeile vorgeben. Eine normale Befehlszeile für MAC.COM sieht beispielsweise so aus:

```
A>MAC RIB $AB HZ PX
```

Wenn Sie nun MAC von SID aus starten, müssen Sie dem Programm irgendwie »vormachen«, Sie hätten diese Parameter eingegeben. Dazu dient »I«:

```
#I RIB $AB HZ PX
```

Der normale CCP-Kommandopuffer liegt zwischen 0080 und 00FF hex. Mit »D« können Sie sich nach dem I-Befehl den Pufferspeicher ansehen:

```
#D0080,00FF
```

Eine weitere Verwendung findet das I-Kommando beim R-Befehl, der Dateien in den Speicher einliest.

»L« ruft den integrierten Disassembler auf. Er listet der Reihe nach die 8080-Mnemonics an den gegebenen Speicheradressen:

```
#L0100,0108
  0100  MVI  C,09
  0102  LXI  D,0300
  0105  CALL 0005
  0108  RST  06
  0109
```

Befehle wie »EX AF,AF'«, die nicht zu den 8080-Mnemonics gehören, zeigt der Disassembler als »??= 08«.

Mit »M« kopieren Sie Speicherblöcke. Dazu nennen Sie dem Debugger die erste Quelladresse, die letzte Quelladresse und die Zieladresse. Wollen Sie den Speicherinhalt der Adressen 0100 bis 0A00 hex nach 1100 bis 1A00 hex kopieren, schreiben Sie:

```
#M0100,0A00,1100
```

Diese Monitorroutine von SID ist so konzipiert, daß auch sich überlappende Speicherbereiche korrekt übertragen werden.

Der »P«-Befehl setzt den sogenannten »Pass Counter«. Dieser zählt bei Programmen, die im Einzelschrittmodus abgearbeitet werden, wie häufig ein Programmpunkt aufgerufen wird. Dazu wird ein Zähler von 255 bis 1 herunterzählt. Sobald der Wert 1 erreicht ist, verwandelt sich der Passpoint in einen echten Breakpoint.

#P0140 setzt einen Passpoint an die Adresse 0140 hex. Der Zähler ist 1. #P0140,34 legt auf die Adresse 0140 hex einen Passpoint mit dem Zählerwert 34 hex. #P listet alle verwendeten Passpoints zusammen mit dem jeweiligen Zähler auf. #-P0140 entfernt den Passpoint von der Adresse 0140 hex wieder.

#-P löscht alle Passpoints.

Der »R«-Befehl (Read File) dient dazu, eine mit »I« angewählte Datei in den Speicher einzulesen.

#R liest die Datei direkt in den Speicher.

#R0100 bestimmt einen Versatz von 0100 hex Bytes.

»S« (Set Memory) zeigt den Inhalt von Speicherstellen an und gibt Ihnen Gelegenheit, diesen abzuändern:

```
#S0100
0100 01 FF
0101 03 'A'
0102 02 #255
0103 02 .
```

Hier geben Sie FF, 'A', #255 und den Punkt an. Die übrigen Texte sind Systemmeldungen.

Eine Erweiterung erfährt »Set Memory« durch die Wort-Option. »SW« ändert 16-Bit-Werte ab:

```
#SW0200
0200 3454 FFFF
0202 4331 #32767
0204 3212 'AL'
0206 6431 .
```

FFFF, #32767, 'AL' und der Punkt stammen in diesem Dialog vom Benutzer.

Eine Bearbeitung von Maschinencode-Programmen in Einzelschritten ermöglicht »T« – die Trace-Option.

Wenn Sie »T20« eingeben, arbeitet der Computer 20 hex (32) Programmschritte ab. Sie sehen dann, wie gleichzeitig Texte ausgegeben werden und das Maschinencode-Programm abläuft. Die »Registerleiste« zeigt den Inhalt der CPU-Register, den Wert des Programmzählers und den Maschinencode an der aktuellen Adresse. Sind Bits im Flag-Register gesetzt, erscheinen die Namen der Flag-Bits statt der Bindestriche, zum Beispiel CZ–I. Es gelten folgende Abkürzungen:

C – Carry-Flag
Z – Zero-Flag
M – Minus-Flag
E – Even-Parity-Flag
I = Interdigit-Flag

»T« ohne Parameter arbeitet einen Programmschritt ab und übergibt dann die Kontrolle wieder an den Debugger. »Tn« führt »n« Maschinencode-Befehle aus und ruft danach SID.COM auf.

Eine Spezialform des Tracings stellt »TW« dar, der »Trace without Call«. In diesem Modus, in dem Sie beide T-Optionen anwenden können, wird nur das Hauptprogramm ge»trace«t. Unterprogramme ruft der Computer normal auf, zeigt sie aber nicht an. So ist der Programmfluß besser zu verfolgen (Bild 2).

```
#TW4
----- A=00 B=0000 D=0000 H=
0000 S=0100 P=0100 MVI  C,09
----- A=00 B=0000 D=0000 H=
0000 S=0100 P=0102 LXI  D,0300
----- A=00 B=0009 D=0300 H=
0000 S=0100 P=0105 CALL 0005
-Z--- A=00 B=0009 D=0300 H=
0000 S=0100 P=0108 RST  06
```

»U« (Untrace) arbeitet ähnlich dem Trace-Befehl. Im U-Modus wird aber die Registerleiste nur beim letzten Befehl angezeigt. So zerstört diese Form des Tracings nicht den Bildschirmaufbau. Erlaubt ist »U« (Ausführung eines Befehls), »Un« (Ausführung von n Befehlen), »UW« (ein Befehl, aber kein Tracing von Unterprogrammen) und »UWn« (n Befehle werden ausgeführt, Unterprogramme bleiben aber ausgeklammert).

»X« haben Sie bereits verwendet. Es dient der Anzeige der Registerleiste sowie zur Änderung von Registerinhalten:

#X zeigt die Registerinhalte und den gerade ausgeführten Maschinencode-Befehl an. #XA erlaubt, den Inhalt des Z80-Akkumulators abzuändern. Für die übrigen Register gelten folgende Befehlsaufrufe:

#XB – BC-Register
#XD – DE-Register
#XH – HL-Register
#XS – SP-Register
#XP – Programmzähler

Auch die einzelnen Flag-Bits kann man direkt auslesen und setzen beziehungsweise löschen:

#XC – Carry-Flag
#XZ – Zero-Flag
#XM – Minus-Flag
#XE – Even-Parity-Flag
#XI – Interdigit-Flag

RMAC und MAC produzieren nicht nur HEX- und REL-Dateien, sondern auch SYM-Symboltabellen. Sie können diese zusammen mit dem Objektcode-Programm in SID laden, was die Arbeit sehr vereinfacht.

Die Meldung »SYMBOLS« sagt aus, daß SID.COM die Symboltabelle korrekt geladen hat und nun mit ihr arbeiten kann. »NEXT« ist die erste freie Speicheradresse. Demnach ist der Bereich zwischen 0100 und 0180 hex belegt. PC zeigt den aktuellen Stand des Programmzählers. END gibt die obere Speichergrenze an.

Der Debugger wartet nun auf Ihre Eingaben. Beginnen Sie damit, ihn die vollständige Symboltabelle auflisten zu lassen. Tippen Sie dazu einfach den H-Befehl ohne Parameter ein:



```
#H
0005 BDOS
010E CLS
0009 OUTPUT
0100 TPA
0000 WARM
```

Überall wo Sie bisher nur hexadezimale oder dezimale Konstanten angegeben haben, »versteht« SID.COM jetzt auch Symbole. Machen wir die Probe aufs Exempel: Ab dem Label CLS wurde in CLS.COM der String abgelegt, der unter CP/M Plus den Bildschirm löscht. Er ist vier Byte lang, nämlich ESC/E/ESC/H. »Display Memory« gibt ihn aus:

```
#D.CLS,.CLS+4
010E: 1B 45 1B 48 24      .E.H$
```

Um den Wert eines Symbols zu erhalten, stellen Sie ihm also einen Punkt voran. Weitere Beispiele für die Benutzung der Symbole:

```
#A.TPA
#XP
P=???? .TPA
#D.BDOS
```

Der Disassembler fügt automatisch die Symbole an den richtigen Stellen in den mnemonischen Code ein:

```
#L0100,010D
TPA:
  0100  LXI  SP,0100   .TPA
  0103  MVI  C,09
  0105  LXI  D,010E     .CLS
  0108  CALL 0005       .BDOS
  010B  JMP  0000       .WARM
CLS:
  010E
```

So werden auch Disassembler-Listings gut lesbar.

Falls Sie die Labels allerdings beim Disassemblieren stören, setzen Sie einfach vor »L« ein Minuszeichen. Der Disassembler unterdrückt dann die Symbole.

Neben der »Punkt-Einleitung« (».TPA«, ».BDOS«) versteht SID.COM noch AT ( ), den Klammeraffen, sowie das Gleichheitszeichen »=«.

Das AT-Sign repräsentiert den 16-Bit-Wert, der an der Adresse des Symbols steht. Entspricht TPA dem Wert 0100 hex und steht in 0100 hex die Zahl 3F hex sowie in 0101 hex der Wert 46 hex, so erhalten Sie mit @TPA eben 463F hex. »=» macht das gleiche für Acht-Bit-Zahlen:

```
#H=TPA
003F #63 '?'
```

SID.COM besitzt keinen Befehl, der das Speichern von Daten aus dem Debugger heraus erlaubt. Dazu hielt CP/M 2.2 den residenten Befehl SAVE bereit, der in CP/M Plus fehlt. Ihn ersetzt jetzt auf der Systemdiskette die Datei SAVE.COM. Und so arbeiten Sie damit: Sie starten SAVE vor dem Debugger-Lauf durch Eingabe des Dateinamens ohne Parameter:

```
A>SAVE
```

Jetzt können Sie SID aktivieren:

```
A>SID MAC.COM
```

In der Titelzeile gibt SID unter »NEXT« die Programmlänge an. Das Lowbyte der Zahl ist immer Null. Merken Sie sich diese hexadezimale Zahl. Sie können jetzt am geladenen Programm die gewünschten Änderungen durchführen. Wenn Sie damit fertig sind, drücken Sie CTRL-C, worauf SAVE.COM sich automatisch reaktiviert:

```
CP/M 3 SAVE - Version 3.0
Enter file (type RETURN to exit):
MAC2.COM
```

Hier will SAVE also den Dateinamen des zu speichernden Files wissen. Um SAVE.COM zu verlassen, drücken Sie nur RETURN oder ENTER.

```
Beginning hex address: 0100
Ending hex address:    2F00
```

Das sind die beiden Adreßangaben, die den Anfang und das Ende der Datei angeben. Das Sichern sollte vor jedem Programmstart geschehen, denn man weiß ja nie...

(Martin Kotulla/Udo Reetz)

Deutsche Sprache, schwere Sprache

Deutsche Sonderzeichen unter CP/M auf allen drei Schneider-Computern bietet »Deutscher Zeichensatz«. Die Vortex-Speichererweiterung darf dabei eingebaut bleiben und auch CP/M Plus ist erlaubt.

Deutsche Sonderzeichen unter CP/M 2.2 für den CPC 464 oder den 664 sind schon lange ein »alter Hut«. Anders ist dies unter CP/M Plus auf dem 6128 oder nach Einbau einer Speicherkarte von Vortex. Denn in solch einem Fall ändern sich die Speicheradressen und die meisten vorgestellten Routinen arbeiten dann inkorrekt. »Deutscher Zeichensatz« ist ein universelles Programm für alle Schneider-Computer und fast für jede Ausbaustufe geeignet.

Deutsche Sonderzeichen mit Locomotiv-Basic zu definieren, ist sehr einfach. Der Befehl »SYMBOL AFTER 64« bereitet den Computer auf die Definition der Sonderzeichen vor. Dazu werden die Bitmustertabellen, die für das Aussehen der Zeichen zuständig sind, ins RAM kopiert. Mit SYMBOL AFTER 64 beschränken wir uns dabei auf die nur notwendigen Zeichen mit den Nummern 64 bis 255. Die Zeichensatztabelle belegt dann 1,5 KByte Speicherplatz. Unter CP/M 2.2 gibt es intern eine Entsprechung des SYMBOL AFTER-Befehls. Diese Routine eignet sich aber nicht für unseren Zweck, denn in der TPA steht nirgends genügend Speicherplatz zur Verfügung. Das heißt, nirgendwo ist ein so großes Stück unbelegter und vor Überschreiben geschützter RAM-Bereich vorhanden. »Deutscher Zeichensatz« arbeitet deshalb mit einem Trick. Immer, wenn das Betriebssystem ein anderes Zeichen als ein Steuerzeichen auf dem Bildschirm ausgeben will, wird die Routine »TXT WRITE CHAR« an der Adresse BDD3 hex aufgerufen. An dieser Adresse steht ein Sprungvektor zur eigentlichen Zeichenausgaberoutine. Da dieser Befehl im RAM des Computers steht, kann durch eine Änderung eine »Umleitung« eingeschoben werden. Bevor die eigentliche Zeichenausgaberoutine aufgerufen wird, läuft somit immer ein Vorprogramm ab.

Dieses Vorprogramm testet, ob ein deutscher Umlaut ausgegeben werden soll. Wenn ja, wird die interne Nummer des Umlautes in einen Wert zwischen 247 und 255 umgerechnet. Danach arbeitet das Programm die ursprüngliche Ausgaberoutine ab. Durch diesen Trick braucht die Zeichensatztabelle im RAM nur noch die Zeichen mit den Nummern 247 bis 255 umfassen. Statt 1,5 KByte belegt es nur noch 72 Byte. Von der »Benutzeroberfläche« aus betrachtet (also wenn Sie eine Taste drücken), besitzen die deutschen Umlaute aber immer noch die ursprünglichen Standardnummern.

Unter CP/M 2.2 ist auf allen drei Schneider-Computern der Speicherbereich von BEBF bis BFFF hex dem BIOS-Maschinenstapel vorbehalten. Tatsächlich werden aber nur

eine Handvoll Bytes am Ende dieses Bereiches benutzt. Der Rest am Anfang des Stapels ist groß genug, um die Zeichensatztabelle samt »Ausgabeumleitung« aufzunehmen. Experimente mit dem Disketten-Controller von Vortex zeigen aber, daß – anders als bei dem Controller von Schneider – einige zusätzliche Bytes hinter der Adresse BEBE hex notwendig sind. Damit unsere Routine in allen Ausbaustufen funktioniert, beginnt sie deshalb erst nach einem gewissen Sicherheitsabstand. Das Programm »DEUTSCH.COM« (die CP/M-Routine) besteht aus zwei Teilen. In dem Teil ab BEC8 hex steht die eigentliche Zeichenausgaberoutine. Der andere regelt die Interpretation, ob der Zeichensatz ein- oder ausgeschaltet werden soll, das Übertragen der neuen Ausgaberoutine in den Zielbereich und das korrekte Einbinden ins Betriebssystem. Beim Einbinden werden die versionsabhängigen Adressen angepaßt, so daß das Programm auf allen drei Schneider-Computern läuft. Allein die Konstante »base« legt fest, welche Adresse die Ausgaberoutine endgültig aufnimmt. Wenn Sie den Bereich ab BEBF hex schon anderweitig vergeben haben, können Sie damit die Routine verschieben, beispielsweise in den Sound- oder Kassettenheader-Puffer.

Mit einer Vortex-Speichererweiterung ist der passende RAM-Bereich schwerer zu bestimmen. Die neue Zeichenausgaberoutine wird nämlich nur beachtet, wenn sie im System-RAM des Computers steht. Normalerweise ist in dieser Ausbaustufe unter CP/M-Modus im Bereich von 8000 bis FFFF hex nur das Erweiterungs-RAM sichtbar. Vor der Übertragung der Ausgaberoutine muß also das Erweiterungs-RAM abgeschaltet und hinterher wieder eingeschaltet werden. Das gelingt ganz einfach mit dem Maschinencode »OUT (c),c«. Die »OUT«-Befehle sind so eingesetzt, daß es nichts schadet, wenn keine Erweiterung eingebaut oder diese nicht aktiviert ist. Eine spezielle Abfrage auf eine Erweiterung fällt also weg. Auch beim CPC 6128, der serienmäßig 128 KByte RAM besitzt, haben die »OUT«-Befehle keine negativen Auswirkungen.

Damit ein Interrupt »nicht ins Leere« geht, benötigt man eine ganz bestimmte ROM/RAM-Konfiguration. Um Schwierigkeiten zu vermeiden, sind diese, solange das Erweiterungs-RAM abgeschaltet ist, gesperrt. Diese kurze Sperrzeit der Tastaturabfrage fällt nicht auf, da die Initialisierung der neuen Zeichenausgaberoutine nur eine halbe Millisekunde dauert.

»DEUTSCH.COM« ohne Probleme

Wenn Sie sich in der Maschinensprache des Z80 auskennen, benutzen Sie sicher das Assemblerlisting (Listing 1) zur Eingabe. Die Zeilennummern dienen nur der Übersicht und können weggelassen werden.

Haben Sie im Umgang mit Assembler keine Erfahrung, dann geben Sie den Basic-Lader aus Listing 2 ein und speichern ihn auf einer Diskette. Wenn Sie das Programm mit »RUN« starten, wird die Datei »DEUTSCH.COM« auf der Diskette angelegt. Bei der Meldung »Alles O.K.« steht das eigentliche Zeichensatz-Programm dann korrekt auf der Diskette. Unter CP/M schaltet »A>DEUTSCH EIN« und »A>DEUTSCH AUS« den deutschen Zeichensatz ein und aus. Das Programm aus Listing 1 brauchen Sie jetzt nicht mehr. Aber bewahren Sie es für den Fall, daß Sie beim Eintippen einen Fehler gemacht haben, sicherheitshalber noch auf. Wenn schon beim ersten Lauf des Programms die Meldung »Tippfehler in Zeile x« auftritt, müssen Sie diese Zeile ausbessern. Auch wenn das Basic-Programm mit einer Fehlermeldung abbricht, befindet sich eine Datei mit dem Namen »DEUTSCH.COM« auf der Diskette. Diese ist aber nicht einsatzfähig, also löschen. Beachten Sie, daß das Basic-Programm nicht alle Eingabefehler entdeckt. Zu diesen Fehlern zählen, wenn:

- in einer Zeile zwei benachbarte Zahlen ausgetauscht werden (merkt nur Explora)
- eine ganze Zeile vergessen wurde
- zwei aufeinanderfolgende Zeilen vertauscht wurden.

Die Anordnung der Buchstaben auf der Tastatur und das Abbild der Buchstaben auf dem Bildschirm ist völlig unabhängig voneinander. Das Zeichensatzprogramm verändert zwar das Aussehen der Zeichen »@[\]{|}~« in »§ÄÖÜäöü«, deren Anordnung auf der Tastatur wird aber nicht beeinflußt. Das »ß« erreichen Sie auf der normalen ASCII-Tastatur mit »CTRL-2«, das »ä« mit »SHIFT-[«, das »ü« mit »SHIFT-]« und das »ö« mit »@«. Die großen Buchstaben der deutschen Umlaute liegen jeweils in der anderen Ebene der entsprechenden Tasten. Zum Anlegen der deutschen DIN-Tastatur müssen Sie also zusätzlich das Programm »SETUP.COM« (auf der Schneider-Systemdiskette) beziehungsweise »INSTALL.COM« (bei Vortex) bemühen.

Einen Vorschlag für eine Tastenanordnung, die der DIN-Tastatur entspricht, zeigt die Tabelle. Für jede zu ändernde Taste ist die Tastennummer dezimal und hexadezimal und die Belegung dezimal, hexadezimal und im Zeichen gegeben. So können Sie sich die für Ihre System-Routine notwendigen Angaben leicht heraussuchen. Tasten, die nicht geändert werden müssen, sind in dieser Tabelle nicht vermerkt.

Unbelegte Tasten haben übrigens nicht den Code 0, sondern 255. Tasten, die einen Wert zwischen 128 und 159 aufweisen, werden als Funktionstasten interpretiert. So können die Funktionstasten auch auf andere Tasten als den Zehnerblock der Tastatur gelegt werden. Die CLR-Taste besitzt bei uns den Code für »CTRL-H« (Backstep). Unter CP/M können Sie dann diese Taste zum Löschen benutzen. Die DEL-Taste unterstützt nämlich nicht alle Programme. Die ESC-Taste wurde mit dem Code für den Standard-ESC versehen. Nur auf der Taste CTRL-ESC bleibt der ursprüngliche – nicht standardgemäße – Wert erhalten. So lassen sich auch Programme bedienen, die speziell für die Schneider-Computer entwickelt wurden.

Die deutschen Sonderzeichen werden beim Lesen vom Bildschirm und bei der Ausgabe in der Entsprechung des Basic-Zeichenausgabemodus »TAG« nicht erkannt. Unter CP/M kommen diese Routinen aber so gut wie nie zur Anwendung, so daß dies keine Einschränkung darstellt.

(Helmut Tischer/hg)

Tastennummer		Belegung								
		Taste normal			Taste und SHIFT			Taste und CTRL		
Hex	Dez	Hex	Dez	Zeichen	Hex	Dez	Zeichen	Hex	Dez	Zeichen
10	16	08	8	^H(BS)	08	8	^H(BS)	08	8	^H(BS)
11	17	2B	43	+	2A	42	*	FF	255	(frei)
13	19	23	35	#	27	39	'	FF	255	(frei)
16	22	3C	60	<	3E	62	>	FF	255	(frei)
18	24	5E	94	^	60	96	'	1E	30	^^(RS)
19	25	7E	126	^3F	63	?		(unverändert)		
1A	26	7D	125	}	5D	93	]	1D	29	^](GS)
1C	28	7B	123	{	5B	91	[	1B	27	^[(ESC)
1D	29	7C	124	]	5C	92	\	1C	28	^\(FS)
1E	30	2D	45	-	5F	95	__	1F	31	^__(US)
1F	31	(unverändert)			3A	58	:	(unverändert)		
20	32	(unverändert)			3D	61	=	FF	255	(frei)
27	39	(unverändert)			3B	59	;	(unverändert)		
29	41	(unverändert)			2F	47	/	(unverändert)		
2B	43	7A	172	z	5A	90	Z	1A	26	^Z(SUB)
39	57	(unverändert)			40	64	@	00	0	^@(NUL)
41	65	(unverändert)			(unverändert)			FF	255	(frei)
42	66	1B	27	^[(ESC)	1B	27	^[(ESC)	(unverändert)		
47	71	79	171	y	59	89	Y	19	25	^Y(EM)

Tabelle. So legen Sie Ihre DIN-Tastatur an

```
000001  ;Deutscher Zeichensatz unter CP/M 2.2 für Schneider-Computer
000002  ;
000003  ;****** Computertypabhängige Adressen
000004  flg464  equ     4Ah             ;Lo-Byte Bildschirmausgabe 464
000005  mtabl4  equ     0B294h          ;Zeiger auf Usermatrixflags 464
000006  mtabl6  equ     0B734h          ;Usermatrixflags 664/6128
000007  ;
000008  ;****** Adressen sowohl für CPC464 als auch CPC664
000009  txtwrt  equ     0BDD3h          ;TXT WRITE CHAR
000010  oldstk  equ     1000h           ;lokaler Stapelbereich
000011  bdos    equ     0005h
000012  dma     equ     0080h
000013  type    equ     09h
000014  ;
000015  ;****** Endgültige Lage des Programms bestimmen
000016  base    equ     0BEC8h          ;hier unbelegter Speicherplatz
000017          org     0100h
000018  ;
000019  ;****** Ersatz wichtiger Z80-Befehle
000020  ldir    equ     0B0EDh          ;Z80-Befehl 'ldir'
000021  outcc   equ     049EDh          ;Z80-Befehl 'out (c),c'
000022  exx     equ     0D9h            ;Z80-Befehl 'exx'
000023  ;
000024  ;****** Texte und Fehlermeldungen
000025          jmp     entry
000026          db      17,13,10,'Deutscher Zeichensatz für Schneid'
000027          db      'er CPC 464/664/6128 unter CP/M 2.2,',13,10,9
000028          db      'mit und ohne Vortex RAM-Erweiterung,',13,10,9
000029          db      'mit Schneider- oder Vortex-Floppydisc',13,10
000030          db      10,'(c) 3.03.1986 by Helmut Tischer',13,10,9,9
000031          db      ' Asternstraße 40, D-8052 Moosburg',13,10,26
000032  strng1: db      13,10,'Ungültige Eingabe: Erlaubt ist nur '''
000033          db      'DEUTSCH EIN'' und ''DEUTSCH AUS''',13,10,'$'
000034  strng2: db      13,10,'''§ÄÖÜ^äöüß'' Deutscher Zeichensatz '
000035          db      'inaktiv',13,10,'$'
000036  strng3: db      13,10,'''§ÄÖÜ^äöüß'' Deutscher Zeichensatz '
000037          db      'aktiv',13,10,'$'
000038  ;
000039  ;****** Lokalen Stapel initialisieren
000040  entry:  lxi     h,0000h
000041          dad     sp
000042          shld    oldstk-2
000043          lxi     sp,oldstk-2
000044  ;
000045  ;****** Wählen, ob einschalten oder ausschschalten
000046          lxi     d,strng1        ;Fehlermeldung: falsche Eingabe
000047          lxi     h,dma           ;Hier steht die Eingabe
000048          mov     a,m
000049          inx     h
000050          cpi     4               ;1.Byte=Eingabelänge=4
000051          jnz     finis           ;Meldung anzeigen+Programmende
000052          inx     h
000053          mov     a,m
000054          inx     h
000055          cpi     'E'             ;Text 'EIN' erkannt
000056          jz      ein
000057          cpi     'A'             ;Text 'AUS' erkannt
000058          jnz     finis           ;Meldung anzeigen+Programmende
000059  ;
000060  ;****** Deutschen Zeichensatz abschalten
000061  ;(Expansions-RAM im Bereich 8000-FFFF abschalten)
000062          di                      ;Interrupts sperren
000063          db      exx             ;Z80-Befehl 'exx'
000064          mov     e,c             ;bc' enthält Konfiguration
000065          mov     a,c
000066          ani     9Fh             ;ohne Erweiterung wirkungslos
000067          mov     c,a
000068          dw      outcc           ;Z80-Befehl 'out (c),c'
000069          mov     c,e             ;Register c restaurieren
000070          db      exx             ;Z80-Befehl 'exx'
000071  ;(Test, ob Zeichensatz schon ausgeschaltet)
000072          lda     txtwrt+2        ;Hi-Adresse von TXT WRITE CHAR
000073          ani     0c0h            ;Zeigt ins ROM oder RAM?
000074          jz      quitof          ;Zeigt schon ins ROM
000075          lhld    crom4+1+versatz ;Enthält Zeiger auf alte Routine
000076  ;(Eigentliche Zeichensatz-abschaltung)
000077          shld    txtwrt+1        ;Zeiger restaurieren
000078  ;(Alte RAM- und ROM-Konfiguration wiederherstellen)
000079  quitof: db      exx             ;Z80-Befehl 'exx'
000080          dw      outcc           ;Z80-Befehl 'out (c),c'
000081          db      exx             ;Z80-Befehl 'exx'
000082          ei                      ;Interrupts freigeben
000083  ;(Bildschirmmeldung und Programmende)
000084          lxi     d,strng2        ;Meldung: inaktiv
000085          jmp     finis           ;Textausgabe+Warmstart
000086  ;
000087  ;****** Deutschen Zeichensatz einschalten
000088  ;(Expansions-RAM im Bereich 8000-FFFF abschalten)
000089  ein:    di                      ;Interrupts sperren
000090          db      exx             ;Z80-Befehl 'exx'
000091          mov     e,c             ;bc' enthält Konfiguration
000092          mov     a,c
000093          ani     9Fh             ;ohne Erweiterung wirkungslos
000094          mov     c,a
000095          dw      outcc           ;Z80-Befehl 'out (c),c'
```

```
000096          mov     c,e
000097          db      exx                 ;Z80-Befehl 'exx'
000098  ;(Test, welche Computerversion vorliegt)
000099          lhld    txtwrt+1            ;Interne Adresse TXT WRITE CHAR
000100          mov     a,h                 ;Zeigt ins ROM oder ins RAM?
000101          ani     0c0h
000102          jnz     quiton              ;Zeichensatz bereits aktiv
000103          mov     a,l                 ;LO-Adresse = Flag 464/664/6128
000104          lxi     d,mtabl4            ;Usermatrixpointer CPC464
000105          cpi     flg464              ;CPC464?
000106          jz      config
000107          lxi     d,mtabl6            ;Usermatrixpointer CPC664
000108  ;(Zeichensatzprogramm Konfigurieren)
000109  config: shld    crom1+1             ;hl enthält Routinenadresse
000110          shld    crom2+1
000111          shld    crom3+1
000112          shld    crom4+1
000113          xchg                        ;de nethält Zeiger auf
000114          shld    marom1+1            ;Usermatrixflags
000115          shld    marom2+1
000116          inx     h                   ;weitere Flagbytes
000117          inx     h
000118          shld    mbrom1+1
000119          shld    mbrom2+1
000120  ;(neuen Betriebssystemvektor eintragen)
000121          lxi     h,base
000122          shld    txtwrt+1
000123  ;(Verschieben des Programms an endgültige Position
000124          lxi     d,base              ;Endgültige Lage
000125          lxi     h,start             ;jetzige Lage
000126          lxi     b,ende-start        ;Programmlänge
000127          dw      ldir                ;Z80-Befehl 'LDIR'
000128  ;(Alte RAM- und ROM-Konfiguration wiederherstellen)
000129  quiton: db      exx                 ;Z80-Befehl 'exx'
000130          dw      outcc               ;Z80-Befehl 'out (c),c'
000131          db      exx                 ;Z80-Befehl 'exx'
000132          ei                          ;Interrupts freigeben
000133  ;(Bildschirmmeldung und Programmende)
000134          lxi     d,strng3
000135  ;
000136  ;****** Programmende, Rücksprung in CCP
000137  finis:  mvi     c,type              ;Textausgabe
000138          call    bdos
000139          lhld    oldstk-2
000140          sphl
000141          ret
000142  ;
000143  ;************************************************************
000144  ;Hier beginnt der systenresidente Teil des Programm
000145  ;(Umkodieren: deutsche Zeichen in Bereich F7h bis FFh)
000146  start:  cpi     '§'
000147          jz      code40+versatz      ;Codewandlung
000148          cpi     'Ä'
000149  crom1:  jc      $-$                 ;später wird Sprung zu
000150          cpi     '_'                 ;TXT WRITE CHAR eingetragen
000151          jc      code5f+versatz      ;Codewandlung
000152          cpi     'ä'
000153  crom2:  jc      $-$                 ;keine Änderung
000154          cpi     7Fh
000155  crom3:  jnc     $-$                 ;keine Änderung
000156  code7f: adi     0CAh                ;äöüß->Codes FCh-FFh
000157  code40: adi     01Ah                ;§->Code F7h
000158  code5f: adi     09Dh                ;Äöü^->Codes F8h-FBh
000159  ;(selbstdefinierte Matrix einschalten)
000160          push    h
000161          lxi     h,0FFF7h            ;Usermatrix ab Zeichen F7h
000162  marom1: shld    $-$                 ;später wird Adresse eingesetzt
000163          lxi     h,matrix+versatz
000164  mbrom1: shld    $-$                 ;Adresse der Usermatrix
000165          pop     h
000166  ;(normale Zeichenausgabe)
000167  crom4:  call    $-$                 ;Lader trägt hier Adresse ein
000168  ;(Zeichenmatrix ausschalten)
000169          push    h
000170          lxi     h,0000h
000171  marom2: shld    $-$
000172  mbrom2: shld    $-$
000173          pop     h
000174          ret
000175  ;(Zeichensatztabelle für deutsche Zeichen '§ÄÖÜ^äöüß')
000176  matrix: db      03ch,060h,03ch,066h,066h,03ch,006h,03ch
000177          db      0c6h,038h,06ch,0c6h,0feh,0c6h,0c6h,000h
000178          db      0c6h,038h,06ch,0c6h,0c6h,06ch,038h,000h
000179          db      0c6h,000h,0c6h,0c6h,0c6h,0c6h,07ch,000h
000180          db      010h,038h,06ch,0c6h,000h,000h,000h,000h
000181          db      06ch,000h,078h,00ch,07ch,0cch,076h,000h
000182          db      06ch,000h,03ch,066h,066h,066h,03ch,000h
000183          db      06ch,000h,000h,0c6h,0c6h,0c6h,07eh,000h
000184          db      01ch,036h,066h,06ch,066h,066h,0cch,000h
000185  ;
000186  ende:
000187  ;****** Versatz jetzige <-> endgültige Lage
000188  versatz equ     base-start
000189          end
```

Listing 1. Das Assembler-Listing für den »Deutschen Zeichensatz«

```
100 'Basic-Lader fuer die Datei 'DEUTSCH
    .COM'                                   [040A]
110 '                                       [DDB2]
120 OPENOUT"DEUTSCH.COM"                    [5D7C]
130 '                                       [DFB6]
140 z%=10000                                [D404]
150 READ d$:WHILE d$<>"Ende"                [271E]
160 d%=VAL("&"+d$):PRINT#9,CHR$(d%);:p%=
    d%                                      [FE90]
170 FOR i%=1 TO 7                           [6CAA]
180 READ d$:d%=VAL("&"+d$)                  [F29C]
190 PRINT#9,CHR$(d%);:p%=p%+d%              [6A74]
200 NEXT                                    [D0E2]
210 READ q%                                 [B40A]
220 IF p%<>q% THEN PRINT"Tippfehler in Z
    eile "z%:CLOSEOUT:END                   [48C6]
230 z%=z%+10                                [2D78]
240 READ d$:WEND                            [8AC4]
250 CLOSEOUT                                [F74A]
260 PRINT"Alles O.K."                       [2F20]
270 END                                     [AE20]
280 '                                       [0CC2]
10000 DATA C3,84,02,11,0D,0A,44,65, 538   [C092]
10010 DATA 75,74,73,63,68,65,72,20, 798   [C084]
10020 DATA 5A,65,69,63,68,65,6E,73, 825   [FEC6]
10030 DATA 61,74,7A,20,66,7D,72,20, 740   [848E]
10040 DATA 53,63,68,6E,65,69,64,65, 803   [AEA6]
10050 DATA 72,20,43,50,43,20,34,36, 498   [B34A]
10060 DATA 34,2F,36,36,34,2F,36,31, 409   [3496]
10070 DATA 32,38,20,75,6E,74,65,72, 696   [EB9E]
10080 DATA 20,43,50,2F,4D,20,32,2E, 431   [D08E]
10090 DATA 32,2C,0D,0A,09,6D,69,74, 456   [7ADE]
10100 DATA 20,75,6E,64,20,6F,68,6E, 716   [EAC8]
10110 DATA 65,20,56,6F,72,74,65,78, 781   [349E]
10120 DATA 20,52,41,4D,2D,45,72,77, 603   [4C86]
10130 DATA 65,69,74,65,72,75,6E,67, 867   [D8C2]
10140 DATA 2C,0D,0A,09,6D,69,74,20, 438   [F7D0]
10150 DATA 53,63,68,6E,65,69,64,65, 803   [71AA]
10160 DATA 72,2D,20,6F,64,65,72,20, 649   [64A4]
10170 DATA 56,6F,72,74,65,78,2D,46, 763   [9CD0]
10180 DATA 6C,6F,70,70,79,64,69,73, 884   [CEDC]
10190 DATA 63,0D,0A,0A,28,63,29,20, 344   [BE9E]
10200 DATA 33,2E,30,33,2E,31,39,38, 404   [6D7A]
10210 DATA 36,20,62,79,20,48,65,6C, 618   [D484]
10220 DATA 6D,75,74,20,54,69,73,63, 777   [D2A4]
10230 DATA 68,65,72,0D,0A,09,09,20, 392   [D092]
10240 DATA 41,73,74,65,72,6E,73,74, 852   [A49C]
10250 DATA 72,61,7E,65,20,34,30,2C, 614   [BF8A]
10260 DATA 20,44,2D,38,30,35,32,20, 384   [D45E]
10270 DATA 4D,6F,6F,73,62,75,72,67, 846   [E0F4]
10280 DATA 0D,0A,1A,0D,0A,55,6E,67, 370   [A6FA]
10290 DATA 7D,6C,74,69,67,65,20,45, 759   [1DD8]
10300 DATA 69,6E,67,61,62,65,3A,20, 704   [05A0]
10310 DATA 45,72,6C,61,75,62,74,20, 751   [947E]
10320 DATA 69,73,74,20,6E,75,72,20, 741   [0588]
10330 DATA 27,44,45,55,54,53,43,48, 567   [9D76]
10340 DATA 20,45,49,4E,27,20,75,6E, 550   [169E]
10350 DATA 64,20,27,44,45,55,54,53, 560   [3F5E]
10360 DATA 43,48,20,41,55,53,27,0D, 456   [7578]
10370 DATA 0A,24,0D,0A,27,40,5B,5C, 355   [70D4]
10380 DATA 5D,5E,7B,7C,7D,7E,27,20, 756   [6540]
10390 DATA 44,65,75,74,73,63,68,65, 821   [F190]
10400 DATA 72,20,5A,65,69,63,68,65, 746   [F69A]
10410 DATA 6E,73,61,74,7A,20,69,6E, 807   [15D4]
10420 DATA 61,6B,74,69,76,0D,0A,24, 602   [D2B4]
10430 DATA 0D,0A,27,40,5B,5C,5D,5E, 496   [1F12]
10440 DATA 7B,7C,7D,7E,27,20,44,65, 738   [EBFA]
10450 DATA 75,74,73,63,68,65,72,20, 798   [C494]
10460 DATA 5A,65,69,63,68,65,6E,73, 825   [02D6]
10470 DATA 61,74,7A,20,61,6B,74,69, 792   [42BA]
10480 DATA 76,0D,0A,24,21,00,00,39, 267   [9F7A]
10490 DATA 22,FE,0F,31,FE,0F,11,E3, 865   [B73E]
10500 DATA 01,21,80,00,7E,23,FE,04, 581   [2A94]
10510 DATA C2,20,03,23,7E,23,FE,45, 748   [7FD0]
10520 DATA CA,CC,02,FE,41,C2,20,03, 956   [2918]
10530 DATA F3,D9,59,79,E6,9F,4F,ED,1375   [0BA8]
10540 DATA 49,4B,D9,3A,D5,BD,E6,C0,1247   [6B86]
10550 DATA CA,C1,02,2A,F6,BE,22,D4,1121   [9E4C]
10560 DATA BD,D9,ED,49,D9,FB,11,2A,1243   [A0A6]
10570 DATA 02,C3,20,03,F3,D9,59,79, 902   [8FC6]
10580 DATA E6,9F,4F,ED,49,4B,D9,2A,1112   [8E9A]
10590 DATA D4,BD,7C,E6,C0,C2,18,03,1168   [1060]
10600 DATA 7D,11,94,B2,FE,4A,CA,EC,1234   [828A]
10610 DATA 02,11,34,B7,22,32,03,22, 375   [F84A]
10620 DATA 3C,03,22,41,03,22,58,03, 290   [B14E]
10630 DATA EB,22,4E,03,22,5F,03,23, 517   [2ABE]
10640 DATA 23,22,54,03,22,62,03,21, 324   [042A]
10650 DATA C8,BE,22,D4,BD,11,C8,BE,1232   [768C]
10660 DATA 21,2A,03,01,84,00,ED,B0, 624   [12A8]
10670 DATA D9,ED,49,D9,FB,11,58,02,1102   [EF4A]
10680 DATA 0E,09,CD,05,00,2A,FE,0F, 544   [BA28]
10690 DATA F9,C9,FE,40,CA,E3,BE,FE,1641   [F0FC]
10700 DATA 5B,DA,00,00,FE,5F,DA,E5,1105   [EC74]
10710 DATA BE,FE,7B,DA,00,00,FE,7F,1166   [09B2]
10720 DATA D2,00,00,C6,CA,C6,1A,C6,1032   [8422]
10730 DATA 9D,E5,21,F7,FF,22,00,00, 955   [80FE]
10740 DATA 21,04,BF,22,00,00,E1,CD, 692   [97CE]
10750 DATA 00,00,E5,21,00,00,22,00, 296   [C132]
10760 DATA 00,22,00,00,E1,C9,3C,60, 616   [A78E]
10770 DATA 3C,66,66,3C,06,3C,C6,38, 644   [44F6]
10780 DATA 6C,C6,FE,C6,C6,00,C6,38,1210   [8862]
10790 DATA 6C,C6,C6,6C,38,00,C6,00, 866   [4F0E]
10800 DATA C6,C6,C6,C6,7C,00,10,38, 988   [CA0C]
10810 DATA 6C,C6,00,00,00,00,6C,00, 414   [4370]
10820 DATA 78,0C,7C,CC,76,00,6C,00, 686   [A302]
10830 DATA 3C,66,66,66,3C,00,6C,00, 534   [FFB6]
10840 DATA 00,C6,C6,C6,7E,00,1C,36, 802   [2EEA]
10850 DATA 66,6C,66,66,CC,00,00,00, 618   [53B4]
10860 DATA Ende                            [50AA]
```

Listing 2. Mit dem Basic-Lader ist das Zeichensatzprogramm schnell eingegeben

Wordstar de Luxe

Ein bißchen Geschick paßt Wordstar noch besser an Computer und Drucker an. Denn verbesserungsbedürftig ist hier einiges.

Im zweiten Schneider-Sonderheft fanden Sie eine Wordstar-Anpassung für die Schneider-Computer unter CP/M 2.2. Viele Besitzer des CPC 6128, deren Wordstar für CP/M Plus angepaßt ist, schreiben uns, ob es nicht auch für ihre Version eine verbesserte Druckerroutine gibt. Wir haben ihre Anregung aufgenommen und mit »Wordstar de Luxe« eine Anpassung geschrieben, die auf jedem CP/M-Computer eingesetzt werden kann.

Normalerweise ist Wordstar so installiert, daß die Steuerzeichenfolgen zum Hoch- und Tiefstellen von Zeichen nur benutzt werden können, wenn der Drucker einen echten Zeilenvorschub nicht nur vorwärts, sondern auch rückwärts ausführt. Intern kennt Wordstar aber eine Steuerzeichenfolge, die nur einen halben Zeilenvorschub aufruft. Und diese Anweisung versteht nahezu jeder Drucker. Damit können auch Geräte, die nur Papiervorschub kennen, Zeichen hoch- oder tiefgestellt ausdrucken. Mit Hilfe dieser Routine führt Ihr Drucker zwischen zwei Zeilen nicht einen ganzen, sondern zwei halbe Zeilenvorschübe aus. Zwischen diesen beiden Schritten wird (im wahrsten Sinne des Wortes) »zwischen den Zeilen« das Tiefgestellte der vorhergehenden und das Hochgestellte der nächsten Zeile gedruckt. Falls Ihr Drucker keinen differenzierten Zeilenvorschub kennt, stellen Sie einfach die Zeilenhöhe generell auf halbe Höhe und geben bei der Steuerzeichenfolge für normalen doppelten Vorschub an. Besonders bei Ausgabe von Formeln lernen Sie das neue Schriftbild bald schätzen.

Die meisten Drucker arbeiten neben dem Standard-ASCII-Zeichensatz auch mit Grafiksonderzeichen. Häufig findet man darunter griechische Buchstaben oder ähnliches. Normalerweise läßt Wordstar diese nicht zu. Mit der hier gezeigten Anpassung ändert sich das. CTRL-P und dann E gedrückt sagt Wordstar, daß er das nächste Zeichen als Grafikzeichen interpretieren soll. Das normale ASCII-Zeichen übersetzt er dann unmittelbar vor dem Ausdruck in das entsprechende Grafikzeichen. Auf ähnliche Weise kann man mit CTRL-P und R zwischen amerikanischem und deutschem Zeichensatz hin- und herschalten.

Neben diesen besonderen Routinen installiert unsere Anpassung natürlich auch die »gewöhnlichen« Druckarten. Komprimierte Zeichen werden mit CTRL-P A ein- und CTRL-P N ausgeschaltet. Da die meisten Drucker keinen automatischen Farbbandwechsel kennen, wurde die dafür vorgese-

hene Steuerfunktion CTRL-P Y mit Breitschrift belegt. Die beiden restlichen frei definierbaren Steuerzeichen CTRL-P Q und CTRL-Q W regeln das Ein- und Ausschalten der Schönschrift. Doppeldruck, Fettdruck, Unterstreichen, Hervorheben, Überdrucken, Rückwärtsschritt, Phantom-Leerzeichen und Hardspace gibt es auf jedem Drucker, da sie vollkommen hardwareunabhängig gelöst wurden.

Alle hier gezeigten Änderungen beruhen auf der Anpassung bestimmter Teile der Datei »WS.COM«. Die meisten neuen Funktionen werden einfach durch Verändern bestimmter reservierter Steuerzeichentabellen und Markierungen installiert. Schwieriger ist die Zeichensatzwahl und die Ausgabe von Grafikzeichen. Anstelle der reservierten Steuerzeichentabellen für die Textmarkierungen CTRL-P E und CTRL-P R haben wir normalerweise auf dem Drucker nicht definierte Pseudosteuerzeichen eingesetzt. Zusätzlich wurde die Druckroutine durch eine neue ersetzt. Soll keines der Pseudosteuerzeichen ausgegeben werden, arbeitet die neue Routine wie das Original. Tritt aber eines der Pseudosteuerzeichen auf, wird dieses nicht ausgegeben, sondern zu einer ganzen Steuerzeichenfolge übersetzt. Das Resultat hängt dabei von verschiedenen Werten bestimmter Variablen innerhalb der Routine ab (zum Beispiel ob der deutsche Zeichensatz ein- oder ausgeschaltet wird).

CP/M besitzt eine Betriebssystemroutine, die testet, ob der Drucker gerade beschäftigt ist oder ein Zeichen übernehmen kann. Wenn man diese Routine nur dann aufruft, wenn der Drucker wirklich bereit ist, gibt es keine Probleme. Im anderen Fall passiert es häufig, daß sich der Computer aufhängt. Leider ist diese Routine von der Betriebssystemversion abhängig. In Wordstar wird sie deshalb nur unterstützt, wenn sie wirklich vorhanden ist. Und das ist sie standardmäßig nicht. Da Sie Ihre Betriebssystemversion nur äußerst selten wechseln, haben wir diese Routine gleich miteingebunden.

Wordstar kennt nicht nur eine Druckerausgaberoutine, sondern vier. Das Programm wird deshalb so installiert, daß automatisch diejenige aufgerufen wird, die wir angepaßt haben. Das ist die, die mit dem CP/M-LST:-Kanal arbeitet. Eine entsprechende Meldung auf dem Bildschirm ist auch vorgesehen. Der Platz der anderen drei Routinen wird nicht mehr benutzt und steht für die neuen Routinen zur Verfügung.

Anpassung leicht gemacht

Das folgende Programm kann auf jedem beliebigen CP/M 2.2-Computer benutzt werden. Für CP/M Plus muß eine kleine Änderung eingefügt werden. Die Druckersteuerzeichen sind fast alle für Epson-kompatible Drucker angepaßt. Ganz speziell sind sie aber für die Schneider-Drucker »NLQ401«, »DMP 2000« und »Seikosha SP1000CPC« geeignet. Weiter unten finden Sie die Anpassung an andere Drucker.

Als erstes müssen Sie mit einem beliebigen Texteditor das Assemblerlisting (Listing 1) eingeben. Das Programm muß zum Übersetzen unter dem Namen »WSDRUCK.ASM« gespeichert sein. Wenn Sie das Programm mit Ihrem Wordstar eintippen, wählen Sie bitte den Menüpunkt »N« (für »Programmeingabe – Non-Dokument-Modus«). Unter »D« (Texteingabe) fügt Wordstar nämlich unsichtbare Steuerzeichen in den Text ein, die später den Assembler durcheinanderbringen würden.

Noch ein paar Tips, die die Eingabe erleichtern. Dem von uns verwendeten Assembler ist es völlig gleichgültig, ob Sie den Quellcode mit Groß- oder Kleinbuchstaben eingeben. Sogar in einem einzigen Wort dürfen Sie beide beliebig mischen. Bei der Eingabe brauchen Sie also auf solche Kleinigkeiten nicht achten.

```
000001  ;Wordstar besonders gut an verschiedene Drucker anpassen
000002  ;  (Version 22.4.86)
000003  ;
000004  ;   - nur geeignet für Drucker, die über den gewöhnlichen
000005  ;     CP/M-'LST:'-Kanal gesteuert werden können
000006  ;   - nur geeignet für Wordstar, Version 3.0, CP/M-80
000007  ;   - aber für beliebige 8080- oder Z80-Computer
000008  ;     und beliebige WS-Installationen
000009  ;     (z.B. Schneider CPC464/664/6128, Commodore C128)
000010  ;   - Hier aber Voreinstellungen ähnlich Epson-Drucker
000011  ;     (insbesondere Schneider NLQ401, Schneider DMP2000
000012  ;      und Seikosha SP1000CPC)
000013  ;
000014  ;Diese Datei mit 'ASM' assemblieren und die HEX-Datei (nicht COM)
000015  ;  mit Hilfe von 'DDT' und 'I'-Befehl der Datei WS.COM überlagern
000016  ;
000017  ;
000018  ;****** Hier die CP/M-Version eintragen
000019          false   equ     0
000020          true    equ     not false
000021          cpmplu  equ     false   ;für CP/M PLus 'true' eintragen
000022  ;
000023  ;****** Wie hoch soll eine Zeile sein (nur gerade Zahlen erlaubt)
000024  ;  (Angabe in der kleinsten Einheit, bei NLQ401 z.B.: 1/216 Zoll
000025          hoehe   equ     36      ;6 Zeilen pro Zoll
000026  ;
000027  ;****** Einschaltmeldung über Drucker (Zeichenanzahl nicht verändern)
000028          org     01B3h
000029          db      '         Schneider NLQ401          ',15,0
000030          db      '    No communications protocol     ',15,0
000031          db      '        Primary list device        ',15,0,26
000032  ;
000033  ;****** Grundeinstellungen für den Drucker
000034          org     0690h
000035          db      0FFh            ;Drucker kann kein Backspace
000036          db      2               ;Anzahl Anschläge für Fettdruck
000037          db      2               ;Anzahl Anschläge für Doppeldruck
000038  ;
000039  ;****** verschiedene druckerabhängige Steuerzeichenfolgen
000040  ;  (jeder Tabellensatz besteht aus einem Byte für die Länge
000041  ;   gefolgt von den eigentlichen Steuercodes)
000042          org     0696h           ;Wagenrücklauf und Zeilenvorschub
000043          db      2,13,10
000044          org     06A1h           ;Wagenrücklauf ohne neue Zeile
000045          db      1,13
000046          org     06A8h           ;ein halber Zeilenvorschub
```

Listing 1. Verbessern Sie Ihren Wordstar

```
000047          db      4,13,27,'J',hoehe/2
000048          org     06AFh           ;Backspace (nur wenn Drucker kann)
000049          db      0
000050          org     06B5h           ;komprimierte Schrift einschalten
000051          db      1,15
000052          org     06BAh           ;komprimierte Schrift ausschalten
000053          db      1,18
000054          org     06BFh           ;Hochstellen ein/Tiefstellen aus
000055          db      0
000056          org     06C4h           ;Tiefstellen ein/Hochstellen aus
000057          db      0
000058          org     06C9h           ;'^PQ', Hier: Schönschrift ein
000059          db      3,27,'x',1
000060          org     06CEh           ;'^PW', Hier: Schönschrift aus
000061          db      3,27,'x',0
000062          org     06D3h           ;'^PE', Hier: Graphikausgabe
000063          db      1,31            ;Pseudosteuerzeichen
000064          org     06D8h           ;'^PR', Hier: deutsch/amerikanisch
000065          db      1,30            ;Pseudosteuerzeichen
000066          org     06DDh           ;rotes Farbband, hier: Breitschrift
000067          db      3,27,'W',1
000068          org     06E2h           ;schwarzes Band, hier: Ende Breit
000069          db      3,27,'W',0
000070          org     06E7h           ;Drucker vorbereiten (vor Druck)
000071          db      6,13,27,'3',hoehe,27,'6'
000072          org     06F8h           ;Drucker abschalten (nach Druck)
000073          db      0
000074          org     070Bh           ;Zeichen zum durch/unterstreichen
000075          db      '-','_'
000076  ;
000077  ;****** Unterprogrammeinsprünge zur Druckereinstellung
000078          org     070Dh
000079          nop                     ;Drucker vorbereiten
000080          nop
000081          ret
000082          nop                     ;Drucker abschalten
000083          nop
000084          ret
000085  ;
000086  ;****** Welche Druckerausgaberoutinen werden verwendet
000087          org     0717h
000088          db      0               ;Ausgabe an CP/M-'LST:'-Kanal
000089          db      0FFh            ;Druckerstatusroutine vorhanden
000090  ;
000091  ;***** Die Druckerausgaberoutinen für CP/M-'LST:'-Kanal
000092          org     071Ah
000093          jmp     busy            ;Sprung zur neuen Busy-Routine
000094          mov     e,a             ;Zeichen an Drucker schicken
000095          mvi     c,5             ;BDOS-Funktionsnummer
000096          call    print           ;normal BDOS-Aufruf, jetzt neu
000097          ora     a
000098          ret
000099          nop                     ;Zeichen vom Drucker holen
000100          stc                     ;nur für manche serielle Drucker
000101          ret                     ;und primitive Schnittstelle nötig
000102  ;
000103  ;****** Dieser Bereich ist nur frei bei Ausgabe an CP/M-'LST:'-Kanal
000104  ;  (wird hier für neue Druckerroutinen zweckentfremdet)
000105          org     0728h
000106  ;(neue Drucker-Busy-Routine, Version für CP/M 2.2)
000107          if      not cpmplu      ;CP/M 2.2 ?
000108  busy    lxi     d,002Ah
000109          call    bios            ;Bios-Aufruf
000110          cma                     ;a ist 255, wenn nicht beschäftigt
000111          rar                     ;carry 0, wenn nicht beschäftigt
000112          ret
000113  bios    lhld    0001h           ;Anfangsadresse des BIOS holen
000114          dad     d               ;relative Routinenadresse addieren
000115          pchl                    ;BIOS-Routine aufrufen
000116          endif
000117  ;(neue Drucker-Busy-Routine, Version für CP/M Plus)
000118          if      cpmplu          ;CP/M Plus ?
000119  busy    lxi     d,regist        ;Anfangsadresse Registerpuffer
000120          mvi     c,50            ;Indirekter BIOS-Aufruf
000121          call    0005h           ;Als BDOS-Routine
000122          lda     regist+1        ;A-Register holen
000123          cma                     ;a ist 255, wenn Drucker bereit
000124          rar                     ;carry 0, wenn Drucker bereit
000125          ret
000126  regist  db      15              ;BIOS-Nummer 'Druckerstatus'
000127          db      0,0,0,0,0,0,0   ;7 Bytes reservieren
000128          endif
000129  ;(neue Druckerausgaberoutine)
000130  print   cpi     30              ;Pseudozeichen für Zeichensatz?
000131          jz      chrset          ;Zeichensatzumschaltung
000132          lxi     h,flag          ;Flag für Graphikmodus
000133          inr     m               ;Graphikmodus zur Probe einschalten
000134          cpi     31              ;Pseudozeichen für Graphikmodus?
000135          rz                      ;ja -> Graphikmodus bleibt ein
000136          dcr     m               ;Graphikmodus korrigieren
000137          jz      0005h           ;jetzt aus ->normaler Druck
000138  ;(Graphikzeichen an Drucker senden)
000139  gramod  dcr     m               ;Graphikmodus testen
000140          mvi     m,0             ;und löschen
000141          push    psw             ;Zeichen und Modus aufheben
```

Listing 1. Verbessern Sie Ihren Wordstar (Fortsetzung)

```
000142            mvi     e,27           ;Escape an Drucker
000143            call    sende          ;ausgeben
000144            mvi     e,'='          ;Graphikzeichensatz einschalten
000145            call    sende
000146            pop     psw            ;Graphikzeichen und Modus holen
000147            jz      notran         ;keine Übersetzung
000148            ani     1Fh            ;in Bereich 80h bis 9Fh
000149            jnz     notran         ;besondere Behandlung für Code FFh
000150            dcr     a
000151    notran  mov     e,a
000152            call    sende          ;Ausgeben
000153            mvi     e,0            ;Graphikmodus löschen
000154            jmp     sende
000155    ;(deutschen/amerikanischen Zeichensatz einschalten
000156    chrset  mvi     e,27           ;Escape-Code
000157            call    sende          ;Escape an Drucker
000158            lxi     h,flag+1       ;hier steht Zeichensatznummer
000159            mov     a,m
000160            xri     01h            ;bit 0 invertieren
000161            mov     m,a            ;neue Zeichensatznummer
000162            mov     e,a            ;ins e-Register
000163    ;(Byte an Drucker ausgeben)
000164    sende   mvi     c,5            ;BDOS-Funktion drucken
000165            jmp     0005h          ;BDOS-Aufruf
000166    ;(verschiedene Flags)
000167    flag:   db      0              ;am Anfang kein Graphikmodus
000168            db      '6'            ;Flag für deutschen Zeichensatz
000169    ;
000170    ;****** Wie wird Kommunikation mit Drucker abgewickelt?
000171            org     0786h
000172            db      0              ;kein ETX/ACK oder XON/XOFF-Protokoll
000173            db      127            ;Druckerpuffer (hier bedeutungslos)
000174    ;
000175            end
```

Listing 1. Verbessern Sie Ihren Wordstar (Schluß)

Alles was in einer Zeile hinter einem Strichpunkt steht, gilt als Kommentar und dient allein der Übersichtlichkeit. Sie dürfen die Kommentare nach eigenem Ermessen weglassen oder dazusetzen. Ebenfalls nur der Überschaubarkeit wegen stehen die Zeilennummern voran. Auch diese dürfen ersatzlos gestrichen werden.

An den Stellen, an denen ein oder mehrere Leerzeichen stehen, dürfen sich beliebig viele Leerzeichen (und auch Tabulatoren) befinden. Die Anzahl der Leerzeichen sind als rein optische Hilfe anzusehen. Das gilt nicht für Leerzeichen zwischen Anführungszeichen. Dabei handelt es sich um Texte, die unverändert ins Programm übernommen werden müssen. Hier können Sie zwar den Text nach eigenem Gutdünken abändern, aber die Gesamtzahl der Zeichen muß erhalten bleiben. Keinesfalls dürfen Sie innerhalb von Anführungszeichen Leerzeichen durch Drücken der »TAB«-Taste einfügen.

Alles folgende ist jedoch von entscheidender Wichtigkeit. Im Gegensatz zu Basic wirkt sich auch nur eine einzige falsche Zeile fatal aus. Deshalb sollten Sie das Listing zur Sicherheit noch einmal genau überprüfen.

Ein Abschnitt des Programms heißt »Betriebssystemversion anpassen«. Wenn Ihr Wordstar nicht unter CP/M 2.2, sondern unter CP/M Plus arbeitet (also für den 6128 gedacht ist), ersetzen Sie die Zeile »cpmplu equ false« durch »cpmplu equ true«.

Um unser Quellprogramm dem Computer verständlich zu machen, brauchen Sie die Programme »ASM.COM« und »DDT.COM« von Ihrer CP/M-Systemdiskette – und natürlich den gerade eingetippten Text »WSDRUCK.ASM« sowie »WS.COM«, »WSMSGS.OVR« und »WSOVLY1.OVR«. Um die Originale zu schützen, kopieren Sie am besten alle Programme auf eine leere Diskette.

Als nächstes ist »WSDRUCK.ASM« zu assemblieren. Das geschieht einfach durch Eingabe von »ASM WSDRUCK. AAZ«. Beachten Sie, daß die etwas seltsame Zeichenfolge »AAZ« im Namen kein Druckfehler ist, sondern den Assembler steuert. Während des Ablaufs von »ASM.COM« darf außer einigen Angaben über den verbrauchten Speicherplatz keine Meldung auf dem Bildschirm erscheinen. Sobald auf dem Bildschirm irgendeine Zeile aus der Datei »WSDRUCK.ASM« ausgegeben wird, die Sie selbst eingetippt haben (eventuell mit einigen Zusätzen), ist diese Zeile fehlerhaft. In diesem Fall starten Sie den Editor noch einmal und korrigieren das Programm. Beachten Sie aber, daß es sich bei dem angezeigten Fehler auch um einen Folgefehler handeln kann. Untersuchen Sie also immer das ganze Programm. Wenn alles geklappt hat, befindet sich auf der Diskette jetzt eine weitere Datei mit dem Namen »WSDRUCK. HEX«.

Als nächstes rufen Sie DDT durch »DDT WS.COM« auf. Wenn kein Fragezeichen erscheint, ist alles in Ordnung. Taucht ein Fragezeichen auf, dann prüfen Sie, ob die Diskette alle notwendigen Dateien enthält. Ein zweiter Versuch muß dann erfolgreich sein.

Erscheint ein Bindestrich, geben Sie »IWSDRUCK.HEX« und danach »R« ein. Wenn nach »R« das Diskettenlaufwerk anläuft und kein Fragezeichen zu sehen ist, ist alles in Ordnung. Im anderen Falle müssen Sie das Programm mit CTRL-C abbrechen. Bei diesem Befehl ist eine häufige Fehlerursache ein fehlendes »ORG« in der Datei »WSDRUCK.ASM« oder eine überhaupt fehlende Datei »WSDRUCK.HEX«. Also bitte darauf achten!

Als nächstes brechen Sie das DDT-Programm mit CTRL-C ab. Dann tippen Sie »SAVE x WS.COM« ein. Der Wert der Zahl x muß aber zuerst noch berechnet werden. Sie erhalten ihn, indem Sie die vierfache Größe der Datei »WS.COM« in KByte nehmen. Wenn Ihre Wordstar-Version also 16 KByte lang ist, geben Sie »SAVE 64 WS.COM« ein, bei 32-Byte-Programmänge »SAVE 128 WS.COM«.

Beachten Sie, daß nach einem mißglückten Versuch, das Wordstar-Programm zu ändern, die Datei »WS.COM« zerstört ist. Damit Sie im Programm »WSDRUCK.ASM« nicht vergeblich nach Fehlern suchen, müssen Sie deshalb nach jedem Versuch das Wordstar-Programm durch eine neue Kopie ersetzen. Änderungen, die Sie schon früher angebracht haben (zum Beispiel um die Bildschirmausgabe zu verbessern), stören natürlich nicht. Als Ausgangsdatei ist ein schon früher veränderter, aber fehlerfreier Wordstar durchaus verwendbar. Wenn Sie Pech haben, hat die frühere Änderung vor der jetzigen Vorrang. Derartige Fälle sind aber sehr selten. Bestimmte Fehler löschen die ganze Wordstar-Diskette, weshalb Sie die neue Version immer auf einer leeren Diskette testen sollten. Nach Ende von »DDT.COM« stürzt auch ein korrektes Wordstar-Programm manchmal ab. Dagegen hilft ein kurzes Aus- und Wiedereinschalten.

Falls in der Testphase das Programm nicht korrekt läuft, finden Sie hier ein paar Hinweise auf mögliche Fehlerursachen.
- Haben Sie im DDT-Programm bei »I« »WSDRUCK.HEX« eingegeben und nicht versehentlich »WSDRUCK.ASM«?
- Befand sich die neueste Version von »WSDRUCK.HEX« auf der Diskette?
- Haben Sie zwischen dem Abbruch von DDT und dem SAVE-Befehl kein anderes Programm gestartet?

```
10000 'WS-Datei erzeugen, die eine Tabel
      le aller Graphikzeichen ausgibt      [703C]
10010 OPENOUT"Drucker2.tst"                [BC16]
10015 PRINT#9,".P00"                       [B9DA]
10020 PRINT#9,CHR$(23)"Unter WORDSTAR mi
      t dem Drucker NLQ401"                [AC2E]
10030 PRINT#9,STRING$(37,"=")              [C718]
10040 PRINT#9                              [EDFC]
10050 PRINT#9,TAB(21)"zur Verf}gung steh
      ende Graphikzeichen"                 [43F0]
10060 PRINT#9,TAB(21)STRING$(37,"=")       [1434]
10070 PRINT#9                              [DB02]
10080 PRINT#9                              [F204]
10090 PRINT#9                              [F906]
10100 PRINT#9,"Druck Eingabe ! Druck Ein
      gabe ! Druck Eingabe ! Druck Einga
      be ! Druck Eingabe"                  [5C86]
10110 PRINT#9,"-----------------+----------
      -----+-----------------+-----------
      ---+-----------------"               [F9DA]
10120 FOR i%=0 TO 31                       [ADBA]
10130 PRINT#9,CHR$(5)CHR$(5)CHR$(i%+64)"
      {5 SPACE}^PE^PE"CHR$(i%+64)" ! ";    [7B98]
10140 PRINT#9,CHR$(5)CHR$(i%+32)"{5 SPAC
      E}^PE"CHR$(i%+32)"{4 SPACE}! ";      [8D52]
10150 PRINT#9,CHR$(5)CHR$(i%+64)"{5 SPAC
      E}^PE"CHR$(i%+64)"{4 SPACE}! ";      [BC68]
10160 IF i%<>31 THEN PRINT#9,CHR$(5)CHR$
      (i%+96)"{5 SPACE}^PE"CHR$(i%+96)"{
      4 SPACE}! ";                         [4D92]
10170 IF i%<8 THEN READ c$:PRINT#9,CHR$(
      18)c$CHR$(18)"{5 SPACE}^PR"c$"^PR"
      ;                                    [9164]
10180 PRINT#9                              [C406]
10190 NEXT                                 [1EB4]
10200 CLOSEOUT                             [2602]
10210 DATA @,[,\,],{,|,~                   [403E]
                                           [AC0E]
                                           [6786]
```

Listing 2. Alle Grafikzeichen im Überblick

- Traten beim Assemblieren von »WSDRUCK.ASM« Fehlermeldungen auf?
- Haben Sie beim Start von DDT den korrekten Namen »WS.COM« angegeben?
- Sind Sie sicher, daß Sie die neueste Version von »WS.COM« getestet haben?
- Stimmt in der Datei »WSDRUCK.ASM« der Wert von »cpmplu«?

Zu Ihrer Übersicht finden Sie im Listing 2 eine Tabelle aller Grafikzeichen. Eingegeben wird das Programm unter Basic. Nach einem Lauf steht auf der Diskette eine ASCII-Datei mit dem Namen »DRUCKER2.TST«, die mit Hilfe von Wordstar ausgedruckt werden kann. Die Ausgabe funktioniert allerdings nur auf Druckern, bei denen die Zeichenfolge »ESC =« den Grafikzeichensatz einschaltet. Das sind mit Sicherheit die Schneider-Drucker NLQ401 und DMP2000 sowie der Seikosha SP1000CPC, aber auch die meisten Epson-kompatiblen Geräte. Ein Versuch kann nichts schaden.

Geben Sie mit Hilfe von Wordstar als nächstes Listing 3 (Test der Schriftarten) ein und drucken Sie es aus.

Der letzte Test betrifft die Druckerstatusabfrage. Schalten Sie dazu den Drucker aus und vergewissern Sie sich, daß auf Ihrem Computer kein Druckerspooler aktiv ist. Dann editieren Sie einen beliebigen Text. Während der Ausgabe sollten Sie die Tastenkombination CTRL-K P drücken und danach einen Dateinamen eingeben. Mit ESC starten Sie dann den Ausdruck der angegebenen Datei. Diese wird ohne einen speziellen Druckerspooler parallel, während der Bearbeitung eines anderen Textes, ausgedruckt. Wenn die Statusabfrage nicht funktioniert, versucht der Computer vergeblich, ein Zeichen auszudrucken und stellt sich »tot«. Ist die Abfrage aber in Ordnung, wird gar nicht erst der Versuch, ein Zeichen auszugeben, unternommen, solange der Drucker zur Übernahme nicht bereit ist. In diesem Fall können Sie genauso problemlos weiterarbeiten, als würde im Hintergrund überhaupt kein Text ausgedruckt.

Funktioniert alles wunschgemäß, so schalten Sie den

```
000001  ;Neuer Text für das Drucker-Menue von Wordstar
000002  ;  (Vers. 22.4.86)
000003  ;
000004  ;Diese Datei mit 'ASM' assemblieren und die HEX-Datei (nicht COM)
000005  ;  mit Hilfe von 'DDT' und 'I' der Datei 'WSMSGS.OVR' überlagern
000006  ;
000007  ;****** Lage des Menues
000008          org     0D3Ch
000009          db      19,19                  ;Kopf des Menues
000010  ;
000011  ;****** Die Texte des Menues
000012  ;  (jede Zeile muß genau 79 Zeichen enthalten und durch ein Byte
000013  ;   mit dem Wert 0Eh abgeschlossen sein.
000014  ;   Ausnahmen:  erste Zeile 66 Zeichen lang, enthält genau einen Tab
000015  ;               letzte Zeile 74 Zeichen lang (7 Zeilen insgesamt)
000016  ;   zur besseren übersicht wird jede Zeile in drei Abschnitte geteilt)
000017          db      '^P KOMMANDO          ',9,'Steuer'
000018          db      'zeichen in den Text einfueg'
000019          db      'en                   ',14
000020          db      'T/V=Hoch/Tief  ein/aus       '
000021          db      'C=Druckpause                 '
000022          db      'Y=Breitschrift ein/aus       ',14
000023          db      'S=Unterstreich. ein/aus      '
000024          db      'B=Schattenschr. ein/aus      '
000025          db      'D=Doppanschlag   ein/aus     ',14
000026          db      'A=komprimierte Schrift       '
000027          db      'N=komprimiert abschalten     '
000028          db      'X=Durchstreichen ein/aus     ',14
000029          db      'O=fester Leerschritt         '
000030          db      'F=Phantom Leerschritt        '
000031          db      'H=Rueckwaertsschritt         ',14
000032          db      'Q=Schoenschrift ein          '
000033          db      'R=deutsch/amerikanisch       '
000034          db      'ENTER=Zeile ueberdrucken     ',14
000035          db      'W=Schoenschrift aus          '
000036          db      'E/^PE=Graphikzeichen folgt   '
000037          db      'Leerschritt=Abbrechen ',14
000038  ;
000039  ;****** Abschluß des Menues:
000040          db      0
000041  ;
000042          end
```

Listing 4. Ihr Wordstar kennt die neuen Befehle auch

Drucker nun ein. Jetzt beginnt der Drucker zu arbeiten.

Wenn die neue Wordstar-Version einwandfrei läuft, brauchen Sie die Dateien »WSDRUCK.ASM« und »WSDRUCK.HEX« nicht mehr. Für den Fall, daß Sie sich aber einmal einen anderen Drucker kaufen, brauchen Sie aber die Assemblerdatei wieder – also an einem sicheren Ort aufheben.

Im Abschnitt »druckerabhängige Steuerzeichenfolgen« können Sie die Steuerzeichen angeben, die zur Wahl einer bestimmten Druckerfunktion notwendig sind. Hinter jeder Zeile, die mit »db« beginnt, steht dabei ein Tabellensatz. Das erste Byte gibt die Anzahl der folgenden Bytes an. Danach folgen die auszugebenden Steuerzeichen. Bei Epson-kompatiblen Druckern stellt man zum Beispiel den Druck in doppelter Breite durch die Drei-Byte-Folge »ESC W SOH« ein. Wie man aus jeder ASCII-Zeichensatztabelle entnehmen kann, besitzt ein »ESC« den Wert 27 und ein »SOH« den Wert 1. Der Buchstabe »W« kann in Anführungszeichen direkt eingesetzt werden. Im entsprechenden Abschnitt der Datei »WSDRUCK.ASM« steht deshalb »db 3,27,'W',1«. Wenn Sie einen anderen Drucker besitzen, können Sie die entsprechenden Werte leicht anpassen. Schwieriger gestaltet sich die Zeichensatz- und Grafikumschaltung. Wenn Sie selbst kein Assemblerspezialist sind, hilft Ihnen sicher gerne ein Spezialist aus einem Club weiter.

Ein neues Druckermenü

Wenn Ihr neuer Wordstar richtig funktioniert, wollen Sie sicher die neuen Druckerroutinen auch im Menü (Aufruf mit CTRL-P) finden. Dazu geben Sie Listing 4 ein. Das weitere Vorgehen entspricht genau dem bei der Anpassung von »WS.COM«. Geändert wird jetzt aber die Datei »WSMSGS.OVR«. Statt »DDT WS.COM« müssen Sie jetzt »DDT WSMSGS.OVR« eingeben und statt »SAVE 64 WS.COM« »SAVE 99 WSMSGS.OVR«. Überlagern müssen Sie damit natürlich die Datei »WSMSGS.HEX«. Dazu tippen Sie »IWSMSGS.HEX« ein. Wenn der erste Versuch nicht klappt, kopieren Sie vor dem nächsten Versuch eine neue Kopie der Datei »WSMSGS.OVR« auf die Arbeitsdiskette. Die Texte zwischen den Anführungszeichen in der Datei »WSMSGS.ASM« können Sie nach Belieben ändern. Beibehalten müssen Sie allerdings die Anzahl der Zeichen zwischen den Anführungszeichen und alles, was außerhalb der Anführungszeichen steht. Ebenfalls verboten sind Tabulatoren innerhalb der Anführungszeichen.

Interessant ist das »Phantom-Leerzeichen« (CTRL-P F). Ausgedruckt sieht es wie ein echtes Leerzeichen aus. Innerhalb von Wordstar wird es aber wie ein normales Zeichen behandelt. Eine nützliche Anwendung enthalten die unterstrichenen Textteile. Leerzeichen zwischen unterstrichenen Worten werden normalerweise nicht mitunterstrichen. Wenn Sie statt dem echten Leerzeichen nun das Phantom-Leerzeichen benutzen, gilt es nicht als Leerzeichen und wird beim Drucken ebenfalls unterstrichen.

```
******* Druckertest *******

Near-Letter-Quality
Schnellschrift
normal Tiefgestellt normal Hochgestellt normal




normal vergrößert normal
normal komprimiert normal
normal Doppeldruck normal
normal Schattenschrift normal
normal Durchgestrichen normal
normal Unterstrichen mit Phantomspace normal
Backstep:           voilà bientôt
Zeichensätze:       Ascii:          '@[\]{|}~'
                    Din:            '§ÄÖÜäöüß'
Graphikzeichen:     Code 81h..9Fh:
                    Code A0h..FEh:
                    Code FFh:
normal
```

Listing 3. Testen Sie die neue Druckerroutine

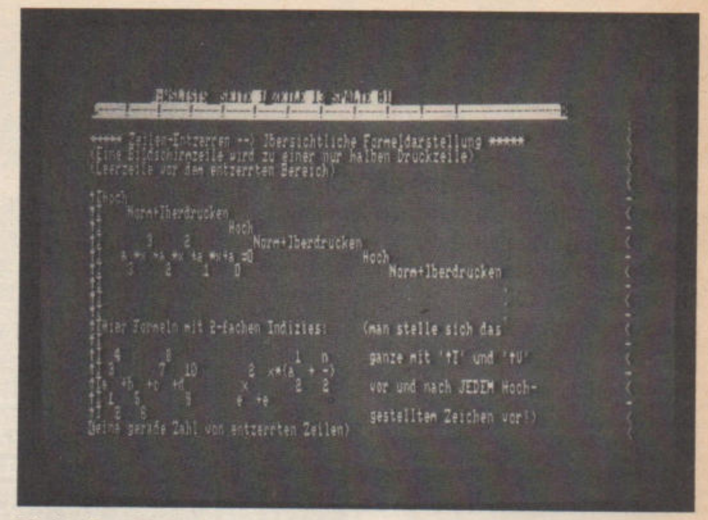


Bild 1. Der Bildschirm-...

```
***** Zeilen-Entzerren --> Übersichtliche Formeldarstellung *****
(Eine Bildschirmzeile wird zu einer nur halben Druckzeile)
(Leerzeile vor dem entzerrten Bereich)

Hoch Norm+Überdrucken Hoch Norm+Überdrucken Hoch Norm+Überdrucken
   a3*x^3+a2*x^2+a1*x+a0=0

Hier Formeln mit 2-fachen Indizies:          (man stelle sich das
                                              ganze mit '^T' und '^V'
                                              vor und nach JEDEM Hoch-
                                              gestelltem Zeichen vor!)
(eine gerade Zahl von entzerrten Zeilen)
```

Bild 2. ...und der Druckertext

Der Rückwärtsschritt funktioniert mit unserem Programm auch bei Druckern, die keinen echten »Backstep« kennen. Mit seiner Hilfe kann man beispielsweise zwei Zeichen übereinanderdrucken und damit Akzente darstellen.

Wenn Sie mit Wordstar viele Formeln mit hoch- und tiefgestellten Indizes eingeben, wirkt die Formel durch die vielen CTRL-V- und CTRL-T-Anweisungen schnell ziemlich zerfleddert. Spätestens dann, wenn Sie zweifache Indizes einsetzen, verlieren Sie endgültig die Übersicht. Das verhindert jedoch ein kleiner Trick. Die Tastenfolge CTRL-P M bewirkt, daß die nächstfolgende Bildschirmzeile die vorhergehende Zeile überdruckt. Auf dem Drucker erfolgt also ein Wagenrücklauf, aber kein Zeilenvorschub. Fügen Sie nun am Anfang und am Ende der ersten Zeile jeweils ein Steuerzeichen CTRL-P T ein, so daß die erste Zeile hochgestellt wird. Auf dem Bildschirm erscheinen zwei vollständig getrennte Zeilen, während auf dem Drucker die Zeilen nur den halben Zeilenabstand aufweisen. Statt am Ende der ersten Zeile darf das zweite CTRL-P T auch am Anfang der zweiten Zeile stehen. Dadurch kommen auf dem Bildschirm untereinanderstehende Spalten auch auf dem Drucker untereinander zu stehen. Die Tastenfolge CTRL-P T, CTRL-P M, CTRL-P T, CTRL-M schafft so einen größeren »entzerrten« Bereich. Wenn Sie jetzt noch mit der TAB-Taste die Zeilen bis zum Rand mit Leerzeichen füllen und den Einfügemodus abschalten, können Sie den Cursor innerhalb des entzerrten Bereiches beliebig bewegen und an beliebigen Stellen Formelzeichen einsetzen. Ohne weitere Unordnung stiftende Steuerzeichen entstehen auf diese Weise die kompliziertesten Formeln mit hoch- und tiefgestellten Indizes. Das Entscheidende ist dabei, daß dadurch die normale Zeilenzählung von Wordstar überhaupt nicht durcheinandergerät. Der Kopf einer Seite erscheint auch bei beliebig vielen entzerrten Bereichen immer ganz oben auf einer Seite. Wie das aussieht, sehen Sie auf Bild 1. Zum Vergleich sehen Sie den zum abgebildeten Text passenden Ausdruck in Bild 2. Sie sehen, ohne großen Aufwand entstehen so perfekte Formeln.

(Helmut Tischer/hg)

Fortsetzung von Seite 51

den Lader mit einem Maschinensprache-Monitor disassemblieren. Dann machen Sie sich auf die Suche nach dem genannten Aufruf. Ihm voraus geht die Zuweisung der Register mit den drei Informationen: Ladeadresse (HL), Länge (DE) und Satzkennzeichen (im Akku). Das Satzkennzeichen dient der Unterscheidung verschiedener Programme. Im Assembler-Quellcode sieht das folgendermaßen aus:

```
LD      HL,xxxx ;Anfangsadresse
LD      DE,xxxx ;Länge
LD      A,xx    ;Satzkennzeichen
CALL    BCA1
```

Benutzen Sie zum Überspielen das Programm aus Listing 5. In Zeile 60 ist die Maschinencode-Routine im String a$ abgelegt. Die Zeilen 70 bis 90 fragen dann nacheinander Anfangsadresse, Länge und Satzkennzeichen ab. Sollte das übertragene Programm zu lang sein (Stichwort: DOS-ROM), lädt unsere Routine es in einen anderen Speicherbereich (denken Sie daran, es später wieder zu »liften«). Sind Bildschirm-Inhalte (beispielsweise Titelbilder) zu übertragen, müssen Sie die Abfragen in den Zeilen 130 bis 150 durch feste Angaben ersetzen, da sie sonst das Bild zerstören. Listing 6 speichert als Beispiel einen Bildschirm in einem Block, den Listing 7 wieder lädt. Durch Änderung der Werte für HL, DE und A sind sie universell für jede Binär-Datei verwendbar. Vergessen Sie aber nicht, daß diese Werte, wie beim Z80 üblich, in der Reihenfolge »niederwertiges Byte, höherwertiges Byte« zu übergeben sind.

Sie kennen nun die wichtigsten Methoden, um Kassettensoftware auf Disketten zu überspielen. Natürlich gilt auch hierfür, daß Sie Kopien nur für den Eigenbedarf anfertigen dürfen.

(Matthias Rosin/Michael Straßer/ja)

Inserentenverzeichnis

Depot-Händler

Tragen Sie Ihre Buchbestellung auf eine Postkarte ein und schicken diese an einen Depothändler in Ihrer Nähe oder an Ihren Buchhändler.

Buchhandlung Herder, Kurfürstendamm 69
1000 Berlin 15, Tel. (030) 8835002, BTX *921782#
Computare Fachbuchhandlung, Keithstraße 18
1000 Berlin 30, Tel. (030) 2139021
Thalia Buchhaus, Große Bleichen 19
2000 Hamburg 36, Tel. (040) 3005050
Boysen + Maasch, Hermannstraße 31
2000 Hamburg 1, Tel. (040) 3005050
Electro-Data, Wilhelm-Heidsiek-Straße 1
2190 Cuxhaven, Tel. (04721) 51288
Buchhandlung Muehlau, Holtenauer Straße 116
2300 Kiel, Tel. (0431) 85085
ECL, Norderstraße 94-96
2390 Flensburg, Tel. (0461) 28181
Buchhandlung Weiland, Königstraße 79
2400 Lübeck, Tel. (0451) 74006-09
Buchhandlung Storm, Langenstraße 10
2800 Bremen 1, Tel. (0421) 321523
Buchhandlung Lohse-Eissing, Marktstraße 38
2940 Wilhelmshaven, Tel. (04421) 41687
Buchhandlung Schmorl u. v. Seefeld, Bahnhofstraße 13
3000 Hannover 1, Tel. (0511) 327651
Buchhandlung Graff, Neue Straße 23
3300 Braunschweig, Tel. (0531) 49271
Deuerlich'sche Buchhandlung, Weender Straße 33
3400 Göttingen, Tel. (0551) 56868
Buchhandlung an der Hochschule, Holländische Straße 22
3500 Kassel, Tel. (0561) 83807
Stern Verlag, Friedrichstraße 24-26
4000 Düsseldorf, Tel. (0211) 373033
Buchhandlung Baedeker, Kettwiger Straße 33-35
4300 Essen 1, Tel. (0201) 221381
Regensberg'sche Buchhandlung, Alter Steinweg 1
4400 Münster, Tel. (0251) 40541-5
Buchhandlung Acker, Johannisstraße 51
4500 Osnabrück, Tel. (0541) 28488
Buchhandlung Brockmeyer, Querenburger Höhe 281/Unicenter
4630 Bochum, Tel. (0234) 701360
Buchhandlung Meier + Weber, Warburger Straße 98
4790 Paderborn, Tel. (05251) 63172
Buchhandlung Phönix GmbH, Oberntorwall 25
4800 Bielefeld 1, Tel. (0521) 58306-38
Buchhandlung Gonski, Neumarkt 24
5000 Köln 1, Tel. (0221) 210528
Mayer'sche Buchhandlung, Ursulinerstraße 17-19
5100 Aachen, Tel. (0241) 4777-136
Buchhandlung Behrendt, Am Hof 5a
5300 Bonn 1, Tel. (0228) 658021
Buchhandlung Cusanus, Schloßstraße 12
5400 Koblenz, Tel. (0261) 36239
Akad. Buchhandlung Interbook, Fleischstraße 61-65
5500 Trier, Tel. (0651) 43596
Buchhandlung W. Finke, Kipdorf 32
5600 Wuppertal 1, Tel. (0202) 454220
Buchhandlung Balogh, Sandstraße 1
5900 Siegen, Tel. (0271) 55298-9
Buchhandlung Naacher, Steinweg 3
6000 Frankfurt 1, Tel. (069) 298050
Buchhandlung Wellnitz, Lautenschlägerstraße 4
6100 Darmstadt, Tel. (06151) 76548
Buchhandlung Feller + Gecks, Friedrichstraße 31
6200 Wiesbaden, Tel. (06121) 304911
Ferber'sche UNI-Buchhandlung, Seltersweg 83
6300 Gießen, Tel. (0641) 12001
Sozialwissenschaftliche Fachbuchhandlung, Friedrichstraße 24
6400 Fulda, Tel. (0661) 75077
Albertis-Hofbuchhandlung, Langstraße 47,
6450 Hanau, Tel. (06181) 24301
Gutenberg Buchhandlung, Große Bleiche 29
6500 Mainz, Tel. (06131) 37011
Buchhandlung Bock + Seip, Futterstraße 2
6600 Saarbrücken, Tel. (0681) 30677
Buchhandlung Wilhelm Hofmann, Bismarckstraße 98
6700 Ludwigshafen, Tel. (0621) 516001
Buchhandlung Loeffler, B 1,5
6800 Mannheim 1, Tel. (0621) 28912
Buchhandlung Stehn, Bahnhofstraße 13
7000 Stuttgart 50, Tel. (0711) 561476
Osiandersche Buchhandlung, Sindelfinger Allee 25
7030 Böblingen
Buchhandlung am Markt, Kramstraße 6
7100 Heilbronn, Tel. (07131) 68682
UNI Buchhandlung Kellner + Moessner, Kaiserstraße 18
7500 Karlsruhe, Tel. (0721) 691436
Osiandersche Buchhandlung, Wilhelmstr. 12
7400 Tübingen, Tel. (07071) 51761
Osiandersche Buchhandlung, Kaiserpassage 8
7410 Reutlingen
Buchhandlung Roth, Hauptstraße 45
7600 Offenburg, Tel. (0781) 22097
Rombach Center, Bertholdstraße 10
7800 Freiburg, Tel. (0761) 49091
Fachbuchhandlung Hofmann, Hirschstraße 4
7900 Ulm, Tel. (0731) 60949
Schauties Elektronik, Bachstraße 52
7980 Ravensburg, Tel. (0751) 26138
Buchhandlung Hugendubel, Marienplatz
8000 München 2, Tel. (089) 2389-1
Computerbücher am Obelisk, Barerstraße 32-34
8000 München 2, Tel. (089) 282383
Pele's Computerbücher, Schillerstraße 17
8000 München 2, Tel. (089) 555229
Universitätsbuchhandlung Lachner, Theresienstraße 43
8000 München 2, Tel. (089) 521340
Buchhandlung Schönhuber, Theresienstraße 6
8070 Ingolstadt, Tel. (0841) 33146/47
Computerstudio Gertrud Friedrich, Ludwigstraße 3
8220 Traunstein, Tel. (0861) 14767
Buchhandlung Pustet, Kl. Exerzierplatz 4
8390 Passau, Tel. (0851) 56945
Buchhandlung Pustet, Gesandtenstraße 6
8400 Regensburg, Tel. (0941)53061
Buchhandlung Dr. Büttner, Adlerstraße 10-12
8500 Nürnberg, Tel. (0911) 232318
Computer-Center-Burger, Leimitzer Straße 11-13
8670 Hof, Tel. (09281) 40075
Sortiments- u. Bahnhofsbuchh. J. Strykowski, Bahnhofplatz 4
8700 Würzburg, Tel. (0931) 54389
Buchhandlung Pustet, Grottenau 4
8900 Augsburg, Tel. (0821) 35437
Kemptener Fachsortiment, Salzstraße 30
8960 Kempten, Tel. (0831) 14413

Schweiz:
Buchhandlung Meissner, Bahnhofstraße 41
5000 Aarau, Tel. (064) 247151
Bücher Balmer, Neugasse 12
6300 Zug, Tel. (042) 214141
Buchhandlung Enge, Bleicherweg 56
8002 Zürich, Tel. (01) 2012078
Buchhandlung Orell Füssli, Pelikanstraße 10
8022 Zürich, Tel. (01) 2118011
Freihofer AG, Wissenschaftliche Buchhandlung, Universitätsstr.11
8033 Zürich, Tel. (01) 3634282
Buchhandlung am Rößlitor, Webergasse 5
9001 St.Gallen, Tel. (071) 228726



Unternehmensbereich Buchverlag
Hans-Pinsel-Straße 2, 8013 Haar bei München

Impressum

Herausgeber: Carl-Franz von Quadt, Otmar Weber

Chefredakteur: Michael Scharfenberger (sc)
Stellv. Chefredakteur: Michael Lang (lg)
Redakteure: Andreas Hagedorn (hg), Thomas Jacobi (ja), Heinrich Lenhardt (hl); Petra Wängler, Eva Hierlmeier (Koordination)
Redaktionsassistenz: Monika Lewandowski (222)

Fotografie: Jens Jancke

Titelgestaltung: Heinz Rauner Grafik-Design

Layout: Leo Eder (Ltg.),
Sigrid Kowalewski (Cheflayouterin)
Rolf Raß, Katja Milles

Produktionsleiter: Klaus Buck (180)

Auslandsrepräsentation:
Schweiz: Markt&Technik Vertriebs AG,
Kollerstrasse 3, CH-6300 Zug,
Tel. (042) 415656, Telex: 862329 mut ch
USA: M&T Publishing Inc., 501 Galveston Dr., Redwood City, CA 94063; Tel. 415-366-3600, Telex 752-351

Manuskripteinsendungen: Manuskripte und Programmlistings werden gerne von der Redaktion angenommen. Sie müssen frei sein von Rechten Dritter. Sollten sie auch an anderer Stelle zur Veröffentlichung oder gewerblichen Nutzung angeboten worden sein, muß dies angegeben werden. Mit der Einsendung von Manuskripten und Listings gibt der Verfasser die Zustimmung zum Abdruck in von der Markt&Technik Verlags AG herausgegebenen Publikationen und zur Vervielfältigung der Programmlistings auf Datenträger. Mit der Einsendung von Bauanleitungen gibt der Einsender die Zustimmung zum Abdruck in von Markt&Technik Verlag AG verlegten Publikationen und dazu, daß Markt&Technik Verlag AG Geräte und Bauteile nach der Bauanleitung herstellen läßt und vertreibt oder durch Dritte vertreiben läßt. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte und Listings wird keine Haftung übernommen.

Anzeigenverkauf: Britta Fiebig (211)

Anzeigenverwaltung und Disposition:
Patricia Schiede (172)

Marketingleiter: Hans Hörl (114)

Vertriebsleiter: Helmut Grünfeldt (189)

Verlagsleiter M&T Buchverlag: Günther Frank (212)

Vertrieb Handelsauflage: Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz: Pegasus Buch- und Zeitschriften-Vertriebs GmbH, Hauptstätter Str. 96, 7000 Stuttgart 1, Tel. (0711) 6483-0

Bezugsmöglichkeiten: Leser-Service: Telefon (089) 4613-249. Bestellungen nimmt der Verlag oder jede Buchhandlung entgegen.

Bezugspreis: Das Einzelheft kostet DM 14,–.

Druck: SOV St. Otto-Verlag GmbH,
Laubanger 23, 8600 Bamberg

Verantwortlich: Für redaktionellen Teil:
Michael Scharfenberger
Für Anzeigen: Ralph Peter Rauchfuß (126).

Redaktionsdirektor: Michael M. Pauly

Vorstand: Carl-Franz von Quadt, Otmar Weber

Anschrift für Verlag, Redaktion, Vertrieb, Anzeigenverwaltung und alle Verantwortlichen:
Markt&Technik Verlag Aktiengesellschaft,
Hans-Pinsel-Straße 2, 8013 Haar bei München,
Telefon (089) 4613-0, Telex 5-22052

Telefon-Durchwahl im Verlag:

Wählen Sie direkt: Per Durchwahl erreichen Sie alle Abteilungen direkt. Sie wählen 089/4613 und dann die Nummer, die in Klammern hinter dem jeweiligen Namen angegeben ist.



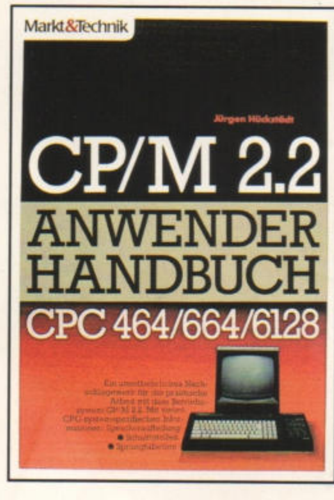
Bücher zu Schneider CPCs

J. Hückstädt
CP/M 2.2 Anwenderhandbuch CPC 464/664/6128
1985, 212 Seiten

Wenn Sie glücklicher Besitzer eines Schneider-Computers sind und mehr wissen wollen über das leistungsstarke Betriebssystem CP/M 2.2, dann ist dieses Buch genau das richtige für Sie! Es behandelt CP/M 2.2 nicht nur in seiner allgemeinen Form, wie sie für sämtliche CP/M-Computer gültig ist, sondern bezieht auch die Hardware der CPC-Computer mit ein.
Best.-Nr. MT 859
ISBN 3-89090-204-9
DM 46,-/sFr. 42,30/öS 358,80

J. Hückstädt
CP/M Plus Anwenderhandbuch CPC 6128
1986, 256 Seiten

Ein unentbehrliches Nachschlagewerk für die praktische Arbeit mit CP/M-Plus und seinen Hilfsprogrammen. Mit zahlreichen Beispielen.
Best.-Nr. MT 90197
ISBN 3-89090-197-2
DM 46,-/sFr. 42,30/öS 358,80





T. Mossakowski/J. Janneck
ROM-Listing CPC 464/664/6128
Februar 1986, 676 Seiten

Dieses Buch enthält in konzentrierter Form umfassende Informationen über den Aufbau Ihres Computers. Um es optimal nutzen zu können, sollte man mit dem Schneider-BASIC vertraut sein und erste Erfahrungen in der Maschinensprache des Z80 besitzen. Zu jeder Routine im Listing sind die Übergabe-Parameter aufgeführt. Verschiedene Tabellen erleichtern das Auffinden einer bestimmten Routine.
Best.-Nr. MT 90134
ISBN 3-89090-134-4
DM 64,-/sFr. 58,90/öS 499,20

Th. Erpel
CPC BASIC-Kurs
1985, 376 Seiten

Ein Buch für den Einstieg in die Bedienung und Programmierung der Schneider-Computer. Alle Beispiele auf Kassette erhältlich. (Best.-Nr. MT 846, DM 29,90*).
Best.-Nr. MT 828
ISBN 3-89090-167-0
DM 46,-/sFr. 42,30/öS 358,80
* inkl. MwSt. Unverbindliche Preisempfehlung











C. Strausch
Schneider CPC Grafik-Programmierung
Januar 1986, 231 Seiten

Dieses Buch wendet sich an die Schneider CPC-Besitzer, die alles über die Grafikfähigkeiten ihres Computers wissen wollen. Es bietet einen umfassenden Überblick über die verschiedenen Anwendungsbereiche der Grafikprogrammierung: zwei- und dreidimensionale Diagrammdarstellungen, Definition und Bewegung von Sprites, Entwurf von Titelgrafiken, Einsatz der Grafik bei der Unterstützung anderer Programme. Alle Beispiele auf Diskette. (Best.-Nr. MT 898, DM 34,90*) und Kassette (Best.-Nr. MT 873, DM 29,90*).
Best.-Nr. MT 90182
ISBN 3-89090-182-4
DM 46,-/sFr. 42,30/öS 358,80
* inkl. MwSt. Unverbindliche Preisempfehlung

J. Hückstädt
Der Schneider CPC 6128
1985, 273 Seiten

Dieses Buch ist für jeden CPC 6128-Besitzer eine wertvolle Hilfe, die vielfachen Möglichkeiten dieses bisher einmaligen Computers kennenzulernen und anzuwenden. Der Computerneuling wird Schritt für Schritt in den Umgang mit dem Computer und in die BASIC-Programmierung eingeführt, bis er alle notwendigen Kenntnisse besitzt, die mancher Profi bereits mitbringt. Aber an dieser Stelle wird das Programmieren mit dem CPC 6128 erst interessant, nämlich dann, wenn es darum geht, eine eigene Dateiverwaltung aufzubauen oder Grafik und Sound zu programmieren. Weiterhin erfahren Sie alles über CP/M Plus auf dem CPC 6128.
Best.-Nr. MT 90192
ISBN 3-89090-192-1
DM 46,-/sFr. 42,30/öS 358,80

C. Strausch
DR-LOGO auf dem Schneider CPC
2. Quartal 1986, ca. 250 S.

Speziell auf die Schneider Computer anwendbar finden Sie in diesem Buch eine strukturierte Anleitung für die praktische Arbeit mit der Programmiersprache LOGO. Mit zahlreichen Beispielen zur Grafik- und Soundprogrammierung. Das letzte Kapitel enthält nützliche Utilities (z.B. SORT-Routinen), viele Informationen über die Aufteilung des Speichers (Speicheranalyse und Tastendefinition), Erklärungen zu den Editorkommandos über die deutschen LOGO-Befehle sowie Lösungsvorschläge zu den Aufgaben.
Best.-Nr. MT 90210
ISBN 3-89090-210-3
DM 46,-/sFr. 42,30/öS 358,80

H. Tischer
Programmentwicklung unter CP/M 2.2 auf dem CPC 464/664
Februar 1986, 340 Seiten

Dieses Buch vermittelt alle Informationen, die zum selbständigen Entwickeln von CP/M 2.2-Programmen nötig sind. Besprochen wird sowohl die grundlegende Funktionsweise des CP/M Betriebssystems als auch alle dem Anwender schon zur Verfügung stehenden Systemroutinen, die diesem viel Arbeit ersparen. Zwei Kapitel beschäftigen sich dabei ausschließlich mit den zusätzlichen Möglichkeiten, die nur die Computer CPC 464/664 bieten.
Kenntnisse der 8080- oder Z80-Assemblersprache sind erforderlich.
Best.-Nr. MT 90209
ISBN 3-89090-209-X
DM 52,-/sFr. 47,80/öS 405,60

C. Strausch
CPC 464 - Programmieren in Maschinensprache
1985, 276 Seiten

Dieses Buch weiht in die Arbeitsweise des BASIC-Interpreters ein und erklärt die Funktionsweise der Bauteile des Geräts und deren Zusammenwirken.
Best.-Nr. MT 829
ISBN 3-89090-166-2
DM 46,-/sFr. 42,30/öS 358,80

Dr. P. Albrecht
MULTIPLAN für den Schneider CPC
1985, 226 Seiten
Best.-Nr. MT 835
ISBN 3-89090-186-7
DM 49,-/sFr. 45,10/öS 382,20

G. Jürgensmeier
WordStar 3.0 mit MailMerge für den Schneider CPC
1985, 435 Seiten

Das unentbehrliche Zusatz-Handbuch für die Arbeit mit dem Schneider CPC.
Best.-Nr. MT 779
ISBN 3-89090-180-8
DM 49,-/sFr. 45,10/öS 382,20

Dr. P. Albrecht
dBASE II für den Schneider CPC
1985, 280 Seiten
Best.-Nr. MT 90188
ISBN 3-89090-188-3
DM 49,-/sFr. 45,10/öS 382,20



Fragen Sie Ihren Buchhändler nach unserem kostenlosen Gesamtverzeichnis mit über 200 aktuellen Computerbüchern und Softwareprogrammen. Oder fordern Sie es direkt beim Verlag an!

Markt & Technik-Fachbücher erhalten Sie bei Ihrem Buchhändler

Bestellungen im Ausland bitte an den Buchhandel oder an untenstehende Adressen.
Schweiz: Markt & Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, ☎ 042/41 56 56
Österreich: Ueberreuter Media Handels- und Verlagsges. mbH, Alser Straße 24, 1091 Wien, ☎ 0222/48 15 38-0

Irrtümer und Änderungen vorbehalten.



Unternehmensbereich Buchverlag
Hans-Pinsel-Straße 2, 8013 Haar bei München



ocean
ULTIMATE
PLAY THE GAME
U.S. GOLD
All American Software
They sold a
MILLION
They sold a
MILLION II
BRUCE LEE
PSION
match point
MATCH DAY
The Hit SQUAD
The Hit SQUAD
STRIKES AGAIN
3 ″-Disk für
Schneider
exclusiv bei
Schneider COMPUTER DIVISION
erhältlich
Vier Spiele-Hits
zum Preis
von EINEM!
Auf Kassette und Diskette
für Commodore 64/128
Schneider CPC
Spectrum 48 K
U.S. Gold Computerspiele GmbH
4044 Kaarst
Distribution in der Schweiz durch HILCU
ENTOMBED REPLACES
KNIGHT LORE ON
THE COMMODORE 64
from...
The Hit SQUAD
MORE FOR LESS
U.S. Gold Computerspiele erhalten Sie in den Fachabteilungen von Kaufhof und Quelle sowie in gutsortierten Computershops